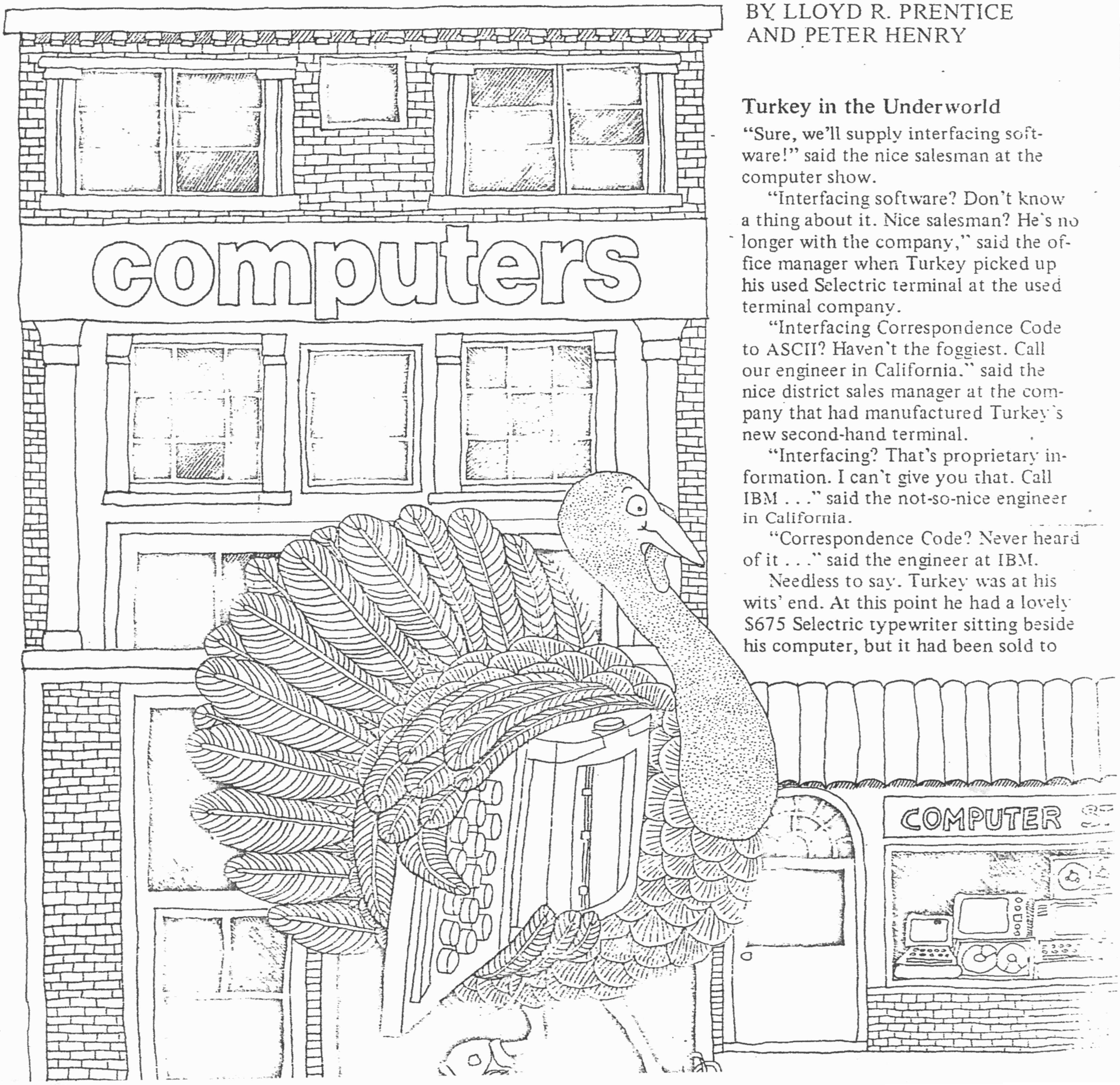
Techno Turkey and his Electric Selectric

BY LLOYD R. PRENTICE
AND PETER HENRY



Turkey in the Underworld

"Sure, we'll supply interfacing software!" said the nice salesman at the computer show.

"Interfacing software? Don't know a thing about it. Nice salesman? He's no longer with the company," said the office manager when Turkey picked up his used Selectric terminal at the used terminal company.

"Interfacing Correspondence Code to ASCII? Haven't the foggiest. Call our engineer in California," said the nice district sales manager at the company that had manufactured Turkey's new second-hand terminal.

"Interfacing? That's proprietary information. I can't give you that. Call IBM . . ." said the not-so-nice engineer in California.

"Correspondence Code? Never heard of it . . ." said the engineer at IBM.

Needless to say, Turkey was at his wits' end. At this point he had a lovely $675 Selectric typewriter sitting beside his computer, but it had been sold to him as a computer terminal — a GTE Information Systems Novar 5-41 terminal, to be exact — and a computer terminal he was determined to have. The only catch was that the Greeks weren't speaking to the Romans, so to speak — that is, his TDL Z80-based computer spoke nice civilized ASCII and the terminal spoke some barbaric tongue called Correspondence. What to do? What to do?

If the high priests of technology couldn't (or wouldn't) help him, Turkey decided, then he had to help himself. But where to start? The first step, he reasoned, was to find a "Rosetta Stone". The ASCII was easy since every computer book ever printed seems to have its obligatory ASCII table in the appendix. But Correspondence Code?

At this point a few write only memory cells decided to read for a change and the name Don Lancaster flashed down the bus to the central processor. Turkey thumbed through *TV Typewriter Cookbook* and behold! There it was! ASCII to Correspondence conversion! But wait . . . It showed Correspondence as a parallel code, and his handy-dandy Novar expects its bits to come marching in one at a time — tra la, tra la. But in what order? And what about baud rate and start and stop bits and all them nice things that Turkey'd tried so feebly to program into his brain during his perusals through the primers?

One fine Saturday (actually it was raining) found Turkey rummaging from floor to ceiling, up one flight and down another, through the labyrinths of MIT's Barker Engineering Library. There were IBM manuals galore. Even a few Correspondence tables popped out of the pages, but these were all of the same stripe as Lancaster's *bon mot* — parallel code with nary a hint about serial communication in the lot.

Oh, despair! Turkey was fully convinced that at least 2,743 computer geniuses on the MIT campus could clear up his troubles without so much as a break in stride toward the coffee machine, but no one seemed remotely interested in talking with a technological turkey like him — the one he approached could bearly bring himself out of his Fourier analysis long enough to give Turkey a blank stare and mutter something about trying to sell a used slide rule.

"I've got to be logical about this," Turkey decided.

He swallowed his pride and dialed the one number he'd vowed by Babbage's beard never to dial again — the computer store that sold him his system. What did he have against the computer store? For one thing, the phone was always busy — a fact that cost him a $7 cab ride on one of those 27 days on which his system was supposed to be ready but wasn't. He later discovered, after he'd put at least one cab driver into an upper income bracket, of course, that they often left the phone off the hook at the computer store so it wouldn't bother them while they were working. Further —

Wait, let's play back the tapes and you can hear for yourself what Turkey was up against.

"Hello, computer store?" (Don't ask me how he got through. It just happened.) "Computer store? Remember me? I bought that $2400 computer system a few months ago? Yes, yes, I'll hold . . ."

The man at the computer store didn't remember Turkey, of course — never did when he called. Turkey waited for the line to stir to life again. It was quite a wait.

"Computer store? Remember that zippy system that was supposed to do word processing and all those other nifty things — you know, the one that was supposed to take three weeks to deliver, but took nearly four months instead?" (Slight pause.)

"No, no, you must have me mixed up with someone else. The name is Turkey . . . T-U-R-K-E-Y." (Another slight pause.)

"Yes, yes, that's me — It's working quite fine, thank you, except the keyboard sticks so I get a screen full of As at the drop of a hat . . ." (Still another slight pause.)

"No kidding! They really wear out in three months? That's amazing! You mean I'll just have to buy a new keyboard every three months, eh? They just don't make things the way they used to, do they? Well, anyway, that's not why I'm calling. . .

"Remember we talked about interfacing a used Selectric terminal before I put down the payment for my computer and you said that it would be no problem? Well I've got the terminal and it's a problem and now I need your help . . ." (And yet another slight pause.)

"You want me to hold while you check with the technician . . ."

At this point Turkey was actually quite heartened, believe it or not, be-

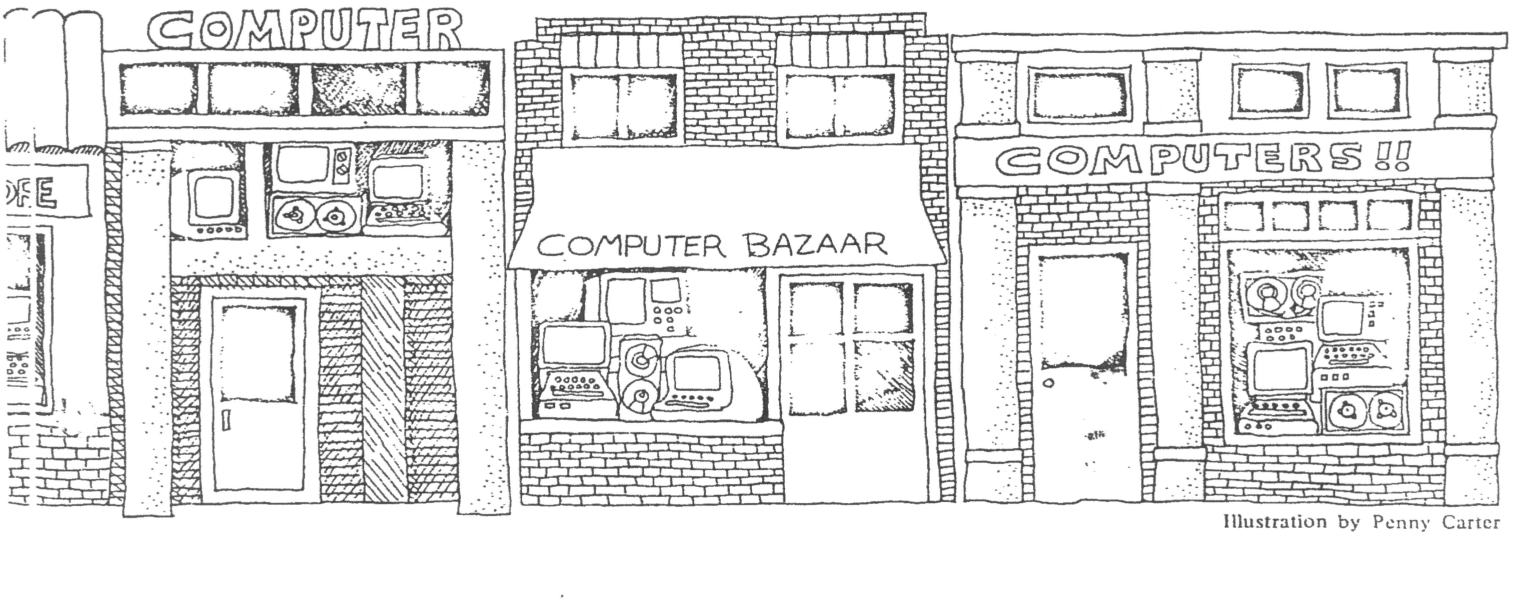


Illustration by Penny Carter

cause the technician was truly a nice fellow. He basically spoke English when he wasn't speaking BASIC. He'd been nice enough to deliver Turkey's system and set it up on his way home from work one night, since Turkey didn't have a car to pick it up at the store himself and had squandered his cab budget on false alarms. If they were all like the technician, Turkey was thinking while the message units ticked over, personal computing could get to be the barrel of monkeys it was cracked up to be.

"Oh yes, hello. The technician says he's not sure if it can be done, and you'd like me to call back in three weeks when you're not so busy . . . Yes, indeed! Thank you, Mr. Computer Store Manager. Thank you Ever so Much!"

Well, back to square one. Turkey was beginning to wonder why Babbage's father had bothered. In fact, Turkey'd plunged to the depths of the technological underworld and they'd torn his mortal frame asunder.

Turkey fights back

And so it came to pass that Turkey was forced into an activity that some folks would say suited him little better than a rented tuxedo – that is, A Little Hard Thinking.

It all started as a few turkey-scratch notes on a coffee-stained Scot towel.

GIVEN 1: Novar 5-41 business data communication terminal designed for serial communication with either another terminal or a computer.

GIVEN 2: Home microcomputer built around the Technical Design Labs ZPU (central processing card based on Zilog Z-80) and System Monitor Board (SMB).

PROBLEM: How does one interface said terminal and computer so that the terminal can be used as a hard copy printer?

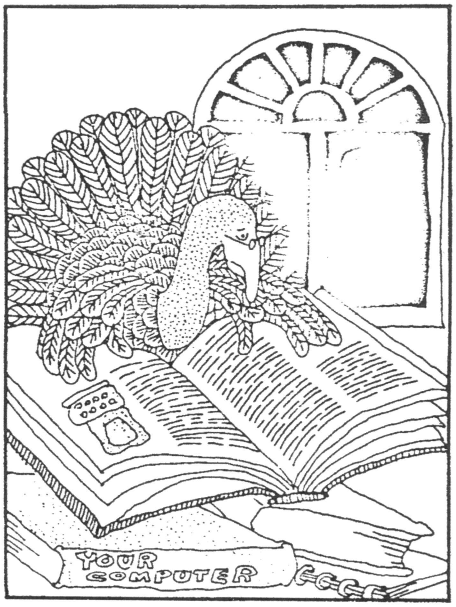
As Turkey thought about it, there were really two different problems here. First, he wasn't quite sure how to hook up the wires between computer and terminal. Indeed, as he looked at the zillion-pin connector sticking out the back of the terminal, he had visions of sparks flying around inside his beloved machines and teleporting him from the roasting pan into the proverbial fire. Second, even if he did hook up the wires correctly, he wasn't sure what kinds of electrical sweet-nothings to tell his computer to whisper down the wire in order to make the terminal sing.

"But," said Turkey, waving a feather in the air, "if I solve the first problem, I might be able to send all kinds of numbers and characters and patterns of bits and things down the wire until I provoke the terminal into chattering back something – anything – and thus I may be able to figure out its accustomed *linqua franca.*

Warming to the task, Turkey set about figuring out the electircal hookup. He had one small clue – RS 232.

"RS 232!" the man at the terminal company had shouted as Turkey'd backed out the door with the terminal in his wings. If he could figure out the meaning of this cryptic phrase, Turkey felt, the hookup problem would fall like the walls of Jericho.

So back to the books went Turkey. Sure enough, his labors paid off. RS 232, he discovered, is an electrical standard for transmission of digital data defined by the Electronic Industries Association. To be more precise, it standardizes the voltage and impedance levels on lines used for data communication. Well, Turkey understood the voltage part okay, but the word "impedance" threw him. After mulling it over a bit, however, he decided to forget about that part of the discussion since he wasn't trying to design circuits anyway, he was only trying to hook up two circuits that were presumably already designed for proper electrical compatibility.



He did find out, however, that there are two versions of RS 232 – B and C. The major difference between the two is the range of voltage swing between the "0" level and the "1" level. With RS 232C the permissible voltage for a logic "1" ranges from -5V to -15V. A logic "0" ranges from +5V to +15V. Voltages between -5V and +5V are in the so-called transition region, which means that the machine currently on the line should not tarry at these never-never-land voltage levels too long or it might confuse the machine on the listening end. RS 232B is quite similar, except the logic "1" is allowed to range from -5V to -25V and the logic "0" from +5V to +25V. It is an older standard.

With this information under his wing, Turkey felt he could measure the voltage swings on the data output pin of the Novar terminal and figure out which version of RS 232 he encountered. But now he had to figure out which pin to measure. Turkey's source of information of RS 232 discussed a 25-pin D connector. Turkey counted the pins on the Novar and, *voila!* – twenty-five! Moreover, the connector was vaguely shaped like a "D". The source further revealed that pin 7 was usually the signal ground, pin 2 the pin dedicated to transmission of data from the terminal and pin 3 the pin for conveying information into the terminal. The only ominous note was that manufacturers didn't always stick to the standard pin designations.

Before slapping a voltmeter on the terminal, Turkey decided to do a little cross-checking. "Never jump to premature conclusions!" he said pompously. When he'd picked up the terminal, the man at the terminal store had thrown in about 10 pages of faintly photocopied logic diagrams. It was a jungle of small-scale integration that was truly quite perplexing, but Turkey looked these up anyway and, sure enough, when he found the input section, it confirmed that pins 2, 3 and 7 were the ones he sought.

By now Turkey felt he was really cooking with methane. He turned on the terminal, whipped out the voltmeter, and measured a very nice -12V across pins 2 and 7. It seemed to Turkey that he had a classical case of RS 232C on his hands.

Now, what about his computer? Turkey looked up the circuit diagram of his System Monitor Board in the manual that TDL had kindly supplied. Since the maximum and minimum supply voltages on the board were +12V and -12V respectively, the RS 232 signal supplied by his computer couldn't possibly swing beyond these extremes, according to Turkey's line of reasoning. Electrically speaking, it looked like he

Pin	Function		
1.	Frame Ground	5.	Clear to send
2.	Transmit	6.	Data set ready
3.	Receive	7.	Signal ground
4.	Request to send	8.	Rec'd Line detect
		20.	Data terminal ready

Figure 1

WIRING DIAGRAM FOR SOLDER SIDE OF 25-PIN RS 232 MALE CONNECTOR GOING TO NOVAR

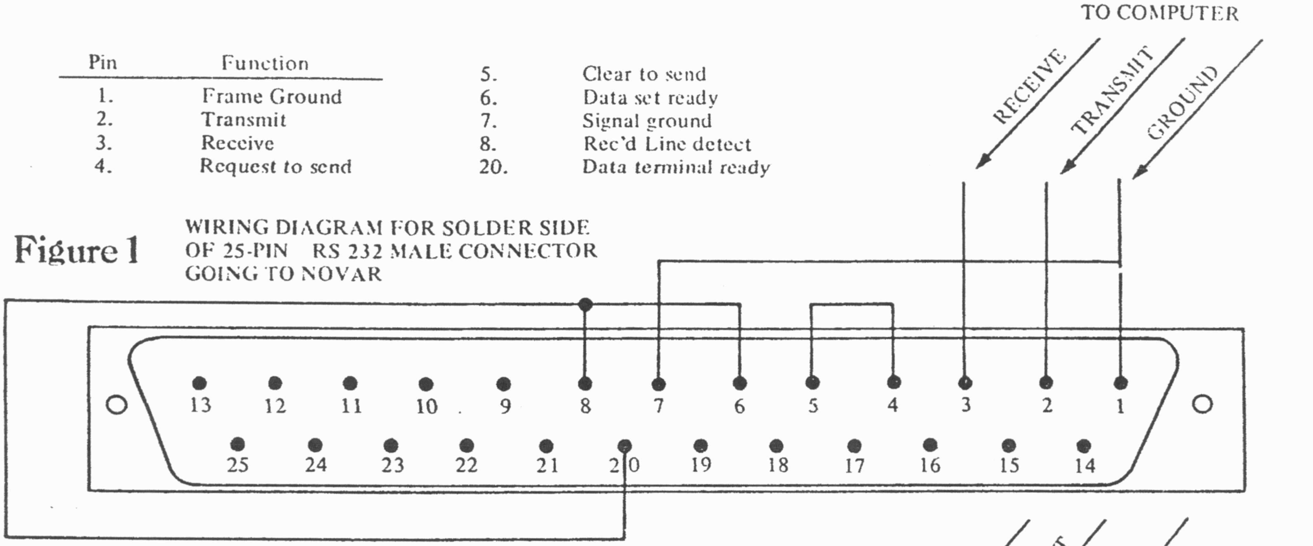
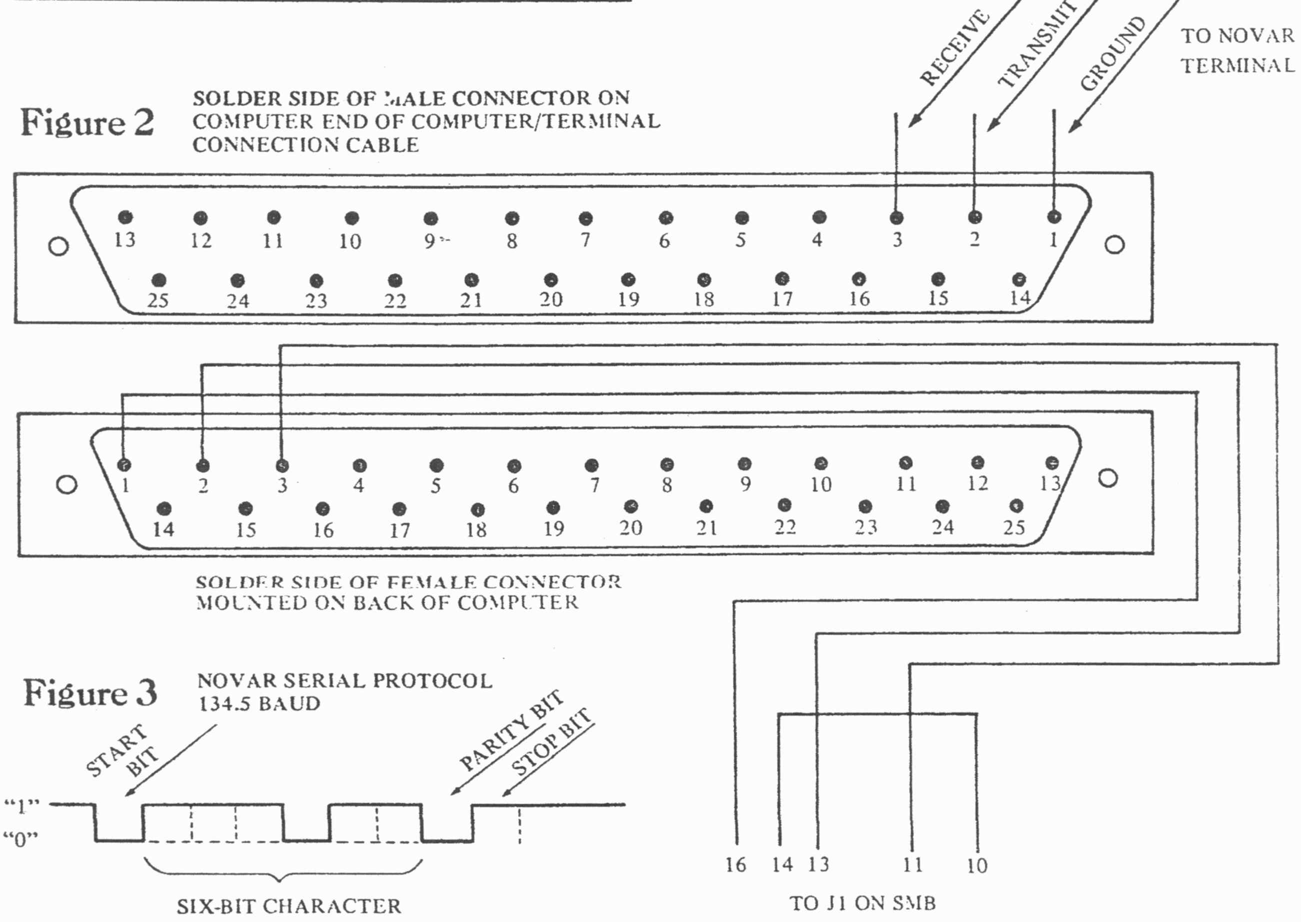


Figure 2

SOLDER SIDE OF MALE CONNECTOR ON COMPUTER END OF COMPUTER/TERMINAL CONNECTION CABLE



SOLDER SIDE OF FEMALE CONNECTOR MOUNTED ON BACK OF COMPUTER

Figure 3

NOVAR SERIAL PROTOCOL
134.5 BAUD

```
FF90  00 3E 00 7C 00 00 00 00 34 2F 00 6E 00 6D 1C 1F
FFA0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
FFB0  00 A0 89 B0 84 88 A8 09 B4 A4 B8 93 3B 37 11 07
FFC0  24 20 10 30 04 08 18 28 38 34 AB 2B B4 13 A4 87
```

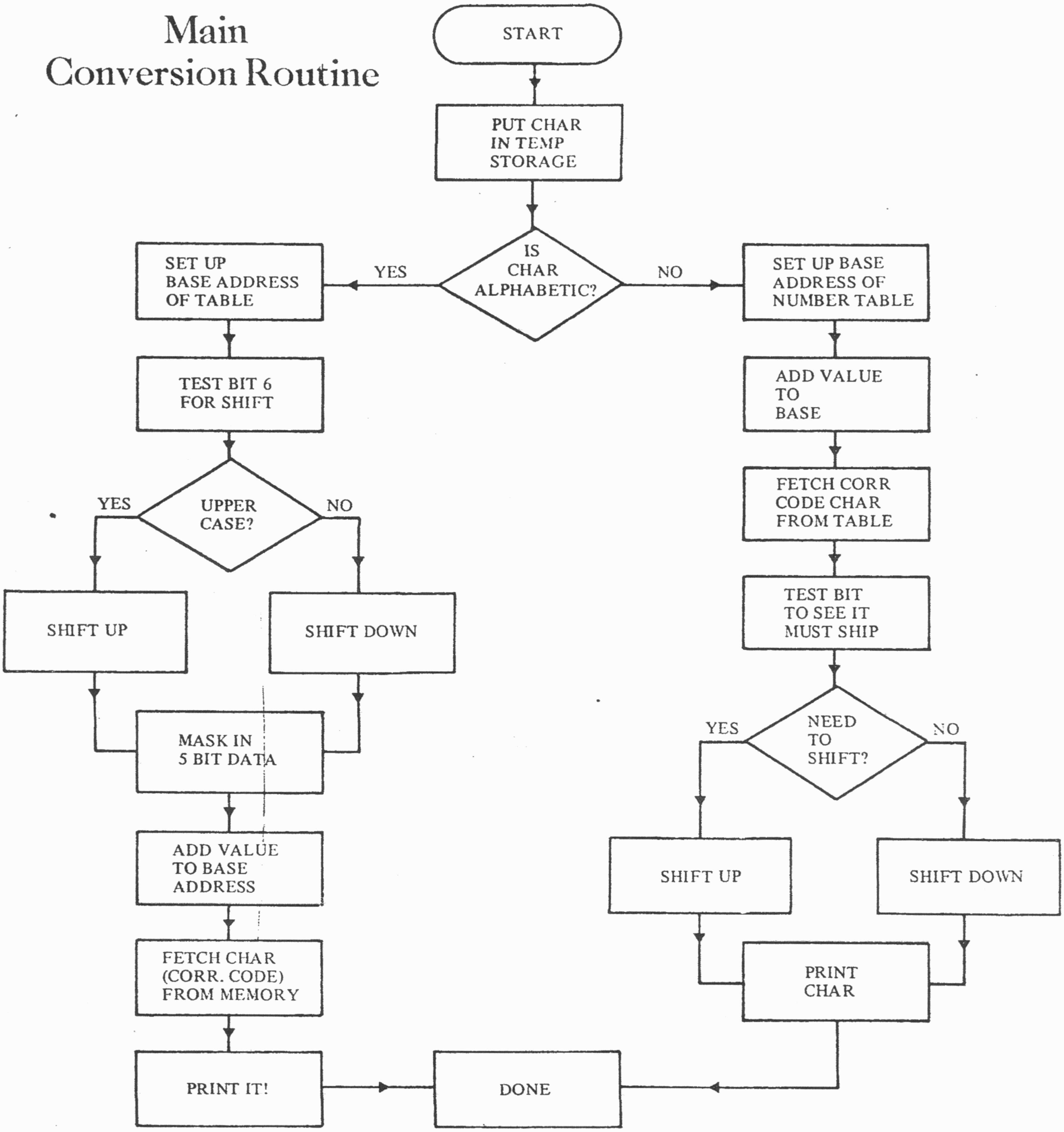
The above is a numeric dump of the 'NONALPH' table, as referred to in the program. The index for this table is 'BASEAD', and the table may be relocated by changing the value of 'BASEAD.'

```
FFD0  90 39 36 3A 2A 0A 33 23 26 19 03 1A 06 21 12 05
FFE0  0B 1B 29 25 02 32 31 35 22 27 14 00 00 00 00 00
```

The above is a numeric dump of the 'ALTAB' table. It may be relocated by changing the value of 'ALTAB' in the driver program, just as with the 'BASEAD' table.

Refer to Software Listing

Main Conversion Routine



Initialization Routine

1. Change vector at F815 to C3 00 FF.
2. Turn on the Novar printer. XMIT, TYPE and READY lights should all be on. Depress TYPE switch. The TYPE light should go out.
3. Using the monitor, enter the following commands:
    a) AL = U   This assigns line printer to user routine.
    b) QO70,15   This initializes ACIA on SMB.
    c) QO71,F4   This initializes the Novar.
    d) QO71,F4   Novar should print out a "9", indicating that initialization is complete.

had the makings of a marriage made in heaven.

Now Turkey looks in the front of his SMB Manual. Here he found very explicit instructions for hooking up a teletype via the RS 232 port. The instructions told him to connect a jumper between pin 4, Request To Send, and pin 5, Clear To Send, on the 25-pin RS 232 connector going to the teletype. It further told him that he should connect a jumper between pin 6, Data Set Ready, pin 8, Receive Line Signal Detect, and pin 20, Data Terminal Ready, and yet another jumper between pin 1, Frame Ground, and pin 7, Signal Ground.

Suddenly, Turkey felt another wave of apprehension. Suppose Novar didn't follow the same pin conventions as the Brand X terminal companies? Turkey went back to the Novar logic diagrams and, with a sigh of relief, discovered that his fears were groundless – pin 4 was indeed Request To Send and the other pins named seemed to be as they should be.

Now Turkey really felt he had it knocked! Ah, false confidence! First off, he sent to some obscure mail order house for an RS 232 25-pin connector set. When the parts arrived he warmed up his trusty old 40-watt soldering iron and tediously soldered the male connector to one end of an appropriate cable and the female connector to the opposite end. Heart racing, he plugged the male end of the cable into the receptacle on the back of the computer and started to plug the female end into his terminal only to discover that he was trying to be the matchmaker for an unnatural relationship! "What's this?" he cried as he realized his foolish oversight. Face red, he apologized to Mdm. Novar and filled out the order form for another male connector.

Ten days later, Turkey tried it again with the proper connectors. When the last solder joint had cooled to a righteous dull haze, he plugged in the cable. Truly excited, he booted up the computer, loaded BASIC, turned on the Novar, and keyed in a short program incorporating lavish LPRINT statements. With remarkable poise, considering how long he had awaited this moment, Turkey punched out RUN on the keyboard. Looking fondly down at the silver golfball in the carriage of the Novar, he hit CR and waited . . . and waited . . . and waited.

Moments later, seriously considering some diabolic scheme of technocide followed by self-immolation, he jerked the plug on his machines, snapped off the lights in the computer room and stomped off to bed. Nothing had happened. Absolutely nothing. Like Mona Lisa, Mdm. Novar had sat, a faintly mocking gleam on her immobile golfball.

Fortunately Turkey isn't the type to hold a grudge for long. The next morning found Turkey hard at work over his much be-cuneiformed Scot towel.

SYMPTOM: Novar just sits there.

POSSIBLE CAUSES: 1) wrong hookup; 2) computer not outputting through proper port; 3) Novar not working; 4) ASCII code incompatible with Novar code.

Somehow Turkey wasn't quite ready to accept the first hypothesis. He reviewed his sources of information and his subsequent reasoning several times over, and each time came up with the same hookup.

Looking fondly down at the silver golfball in the carriage of the Novar, Turkey hit the CR and waited . . . and waited . . . and waited . . .

Hypothesis two he eliminated by checking the output pin from his computer with a voltmeter. The needle of the voltmeter wiggled and waggled just as he'd hoped it would when he ran his little BASIC test program.

Hypothesis three he wasn't quite ready to accept since a nice technician that he'd hired to replace the worn platen on the machine had checked out the Novar's electronics by hooking it into a test program on a time sharing system. The Novar passed with flying colors. While the technician knew nothing about code conversion, he did throw out a few valuable clues.

"Look," he said, "I'm pretty sure it's a six-bit code plus one bit for parity. The way they get the shift characters is by sending a separate character for 'shift up' or 'shift down' such as the case may be. Now electrically, that line just sits there at logic '1' until the computer puts out a character. The first thing that happens is that the line drops to logic '0' for one bit, which is the start bit. This bit tells the terminal that a character is coming down the pike. Then the six-bit character code comes along. The seventh, or parity, bit is either a '1' or a '0' depending upon the number of ones in the character bit. I can't remember if you get a parity '1' if there are an even or an odd number of bits in the character."

"What's a parity and what happens if you get the wrong one?" Turkey asked.

"The parity bit is used for error detection. It tells the terminal that the number of logic ones in a character sent by the computer is the same as the number of logic ones received by the terminal. Nothing much happens, really, if you get the wrong parity," the technician had said. "The transmission error light on the panel next to the keyboard goes on, but otherwise the terminal keeps working."

"That's all there is to it?" Turkey had asked.

"After the parity you need one stop bit at logic '1'. Also, the rate at which the bits get sent down the line is quite important. It has to be 134.5 bits-per-second. Otherwise the terminal won't work properly."

"That's it!" Turkey shouted as he reviewed this conversation in his mind. "I don't know what bit rate my computer is putting out!"

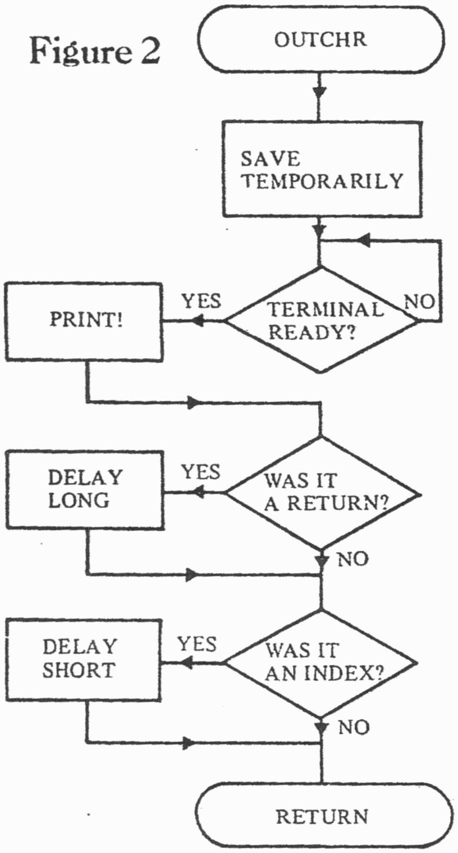
So back to the SMB Manual went Turkey, only to meet up with still another blow. According to the manual, the SMB could generate 9600, 4800, 1200, 600, 300 or 110 baud, but it made no mention of 134.5.

At about this time Turkey was noted by friends to be perpetually morose. He took to walking dark streets alone and frequenting all-night diners. More than once he was seen wearing mismatched socks and the food on his plate often ended up in the disposal.

In truth, Turkey was mulling over complicated schemes of external UARTs and clock generators. At one point he even fantasized a gigantic computer between his TDL system and the Novar that was expressly dedicated to translating ASCII characters into whatever weird code the Novar wanted and putting them out at the proper baud

Conversion Subroutines

Figure 2



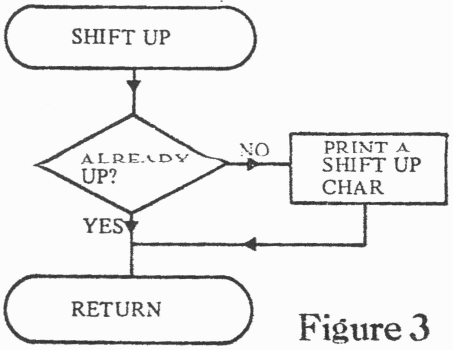


Figure 3

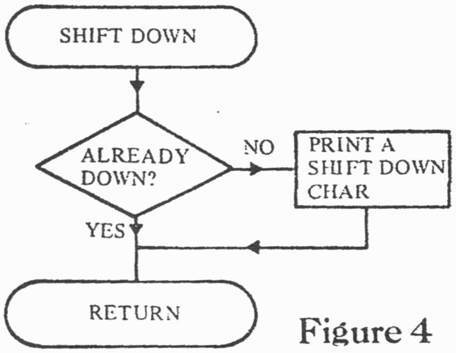


Figure 4

Program Listing

```
                        ;NOVAR TERMINAL DRIVER BY PETER HENRY, 1978
0000'  C5               ENTER: PUSH B
0001'  D5               PUSH D
0002'  E5               PUSH H ;SAVE ALL THE REGISTERS THAT WE WILL USE
0003'  11 D0D0          LXI D,0D0D0H ;SET UP FOR A DELAY
0006'  CD 0083'         CALL COUNT ;AND GO DELAY BEFORE PRINTING
0009'  79               MOV A,C ;PUT IT (CHAR IN THE ACC)
000A'  E640             ANI 040H ; MASK FOR TEST TO SEE IF ALPHABETIC
000C'  281D             JRZ NONALF ;GO TO NONALF IF NOT ALPHABETIC
000E'  21 FFD0          LXI H,ALTAB ;LOAD ALPH TABLE ADDRESS
0011'  79               MOV A,C ;FETCH THE ORIGINAL CHARACTER
0012'  E620             ANI 020H ;GET THE SHIFT BIT ONLY
0014'  2005             JRNZ LCASE ;GO TO LCASE IF LOWER CASE
0016'  CD 0062'         CALL SHFTUP ;SHIFT GOLF BALL UP
0019'  1803             JMPR CONT ;AND CONTINUE WITH OUTPUT
001B'  CD 0071'         LCASE: CALL SHFTDN ;SHIFT GOLF BALL DOWN
001E'  79               CONT: MOV A,C ;FETCH ORIGINAL CHARACTER
001F'  E61F             ANI 01FH ;MASK IN THE DATA ONLY
0021'  1600             MVI D,0H ;SET UP FOR INDEX OF TABLE
0023'  5F               MOV E,A ;.....MORE SET UP
0024'  19               DAD D ;PERFORM THE INDEX UP THE TABLE
0025'  7E               MOV A,M ;FETCH CORR. CODE CHAR.
0026'  CD 004D'         CALL OUTCHR ; AND PRINT IT!!
0029'  181E             JMPR DONE ;AND GO FINISH UP!
002B'  79               NONALF: MOV A,C ;FETCH ORIGINAL CHARACTER
002C'  21 FF90          LXI H,BASEAD ;LOAD NON ALPH. TABLE ADDR IN HL
002F'  E63F             ANI 03FH ;MASK IN THE RAW DATA ONLY
0031'  1600             MVI D,0H ;SET UP FOR THE INDEX
0033'  5F               MOV E,A ;MOVE INDEX VALUE TO E
0034'  19.              DAD D ;PERFORM THE INDEX
0035'  7E               MOV A,M ;FETCH THE CORR. CODE. CHAR.
0036'  C600             ADI 0 ;THIS IS A DUMMY ADD TO SET CONDITIONS
0038'  FA 0040'         JM SHIFTR ;MSB INDICATES A SHIFT UP NEED
003B'  CD 0071'         CALL SHFTDN ;NO UPSHIFT, SO DOWNSHIFT!
003E'  1803             JMPR CONTNU ;AND CONTINUE!
0040'  CD 0062'         SHIFTR: CALL SHFTUP ;SHIFT GOLF BALL UP!
0043'  7E               CONTNU: MOV A,M ;RE-FETCH THE DATA
0044'  E67F             ANI 07FH ;MASK IN THE REAL DATA ONLY!
0046'  CD 004D'         CALL OUTCHR ;PRINT THE CHARACTER,...
0049'  E1               DONE: POP H ;AND WERE ALL DONE!!
004A'  D1               POP D ;RESTORE REGISTERS IN REVERSE ORDER
004B'  C1               POP B ;..........!
004C'  C9               RET ;AND RETURN! SUBRS FOLLOW....
004D'  47               OUTCHR: MOV B,A ;SAVE TEMPORARILY
004E'  DB70             TEST: IN 70H ;GET THE PORT STATUS ***
0050'  CB4F             BIT 1,A ;TEST FOR A READY
0052'  28FA             JRZ TEST ;LOOP IF TERMINAL NOT READY
0054'  78               MOV A,B ;AT LAST!! RESTORE CHAR!
0055'  D371             OUT 71H ;AND PRINT IT!!!
0057'  FE6D             CPI RETURN ;WS IT A RETURN
0059'  CC 0080'         CZ DELAY1 ;IF YES, THEN DELAY LONG
005C'  FE1D             CPI INDEX ;WASIT A LINE FEED?
005E'  CC 008A'         CZ DELAY2 ;IF YES,THEN DELAY SHORT...
0061'  C9               RET ;WERE ALL DONE HERE!!
0062'  3A FFFE          SHFTUP: LDA SHSTAT ;GET THE SHIFT STATUS
0065'  C600             ADI 0H ;DUMMY ADD HERE!!
0067'  F8               RM ;RETURN IF WE ARE ALREADY UP!
0068'  3EFF             MVI A,0FFH ;ELSE...
006A'  32 FFFE          STA SHSTAT ;SET THE STATUS TO UP..
006D'  3E1C             MVI A,SHIFTU ;AND FLIP THE GOLF BALL....
006F'  18DC             JMPR OUTCHR ;BY PRINTING A SHIFT UP! (JMP+RET)
0071'  3A FFFE          SHFTDN: LDA SHSTAT ;SAME AS SHFTUP ROUTINE, EXCEPT...
0074'  C600             ADI 0H ;WE'RE GOING TO SHIFT DOWN!!
0076'  C8               RZ
0077'  3E00             MVI A,0H
0079'  32 FFFE          STA SHSTAT
007C'  3E1F             MVI A,SHIFTD
007E'  18CD             JMPR OUTCHR
0080'  11 0000          DELAY1: LXI D,0H ;THESE ARE JUST DELAY LOOPS,...
0083'  14               COUNT: INR D ;COUNTING UP DE TO 0..........
0084'  20FD             JRNZ COUNT
0086'  1C               INR E
0087'  20FA             JRNZ COUNT
0089'  C9               RET
008A'  11 7070          DELAY2: LXI D,07070H ;SHORT DELAY CONSTANT
008D'  18F4             JMPR COUNT
FFD0                    ALTAB=0FFD0H
FF90                    BASEAD=0FF90H
006D                    RETURN=06DH
001D                    INDEX=01DH
FFFE                    SHSTAT=0FFFEH
001C                    SHIFTU=01CH
001F                    SHIFTD=01FH
```

```
                          +++++ SYMBOL TABLE +++++

ALTAB   FFD0          BASEAD FF90           CONT    001E'          CONTNU 0043'
COUNT   0083'         DELAY1 0080'          DELAY2  008A'          DONE   0049'
ENTER   0000'         INDEX  001D           LCASE   001B'          NONALF 002B'
OUTCHR  004D'         RETURN 006D           SHFTDN  0071'          SHFTUP 0062'
SHIFTD  001F          SHIFTR 0040'          SHIFTU  001C           SHSTAT FFFE
TEST    004E'
```

rate. More typically during this period, Turkey was thinking about giving up the material life altogether for more spiritual pursuits.

Turkey meets the Wizard

One day Turkey found himself browsing listlessly through a strange computer store. A man asked if he needed help and, normally one to keep his problems to himself, Turkey unaccountably found his troubles spilling like a torrent out his beak. The man behind the counter kept nodding his head sympathetically and clucking his tongue at each new episode.

"You got yourself quite a problem there, fellow," said the man when Turkey finished. "I don't think we can help you, but we've got a kid who hangs out here who's a wizard at these problems."

It was all Turkey could do to keep from jumping over the counter and kissing the man.

"His name! His name!" Turkey cried.

"We just call him the Wizard of Newton," the man said as he wrote a name and number on the pad. "He just might be able to give you a hand."

Turkey lost no time looking up the Wizard. At first he was afraid that the computer store man was pulling his leg when he met the fabled magician face-to-face, because the Wizard of Newton was really quite young for your typical card-carrying sorcerer. But the gleam of self-assurance in the Wizard's eye as Turkey explained his problem banished all doubt like bad dreams in the night.

"What kind of chip does your computer use to generate clock signals?" the Wizard asked when Turkey finished.

Now Turkey had never really given this a thought. Indeed, he had to look it up in his SMB Manual.

"A Motorola MC 14411–" said Turkey after studying the schematic.

"I know that chip," said the Wizard. "I'm quite sure it will generate a 134.5 baud clock signal."

After some searching, Turkey found technical information on the MC 14411 in his friend Don Lancaster's *TV Typewriter Cookbook*. Sure enough, pin 14 is supposed to put out a lovely 134.5 clock signal. Now Turkey referred back to his System Monitor Board and discovered a solder connection available at pin 14, but no silkscreen designation of its purpose whatsoever.

"If you connect the TTY Baud Rate jumper on your System Monitor Board to pin 14, then try your test program," said the Wizard, "it just might work."

No sooner said than done and – Behold! The terminal barked to life with a short line of nonsense characters.

Oh, Turkey was happy! He was ecstatic! If only all such lines of nonsense of so little accord could put such joy into our hearts!

"Now all we need," said the Wizard, "is this mysterious Correspondence code, and I can conjure up a conversion program for you faster than you can say Central Processing Unit."

That night Turkey couldn't sleep. "How in the world can I get that code?" he muttered. "On the one hand I have a mechanism that makes a silly golfball dance around like John Travolta. On the other hand I have a stream of logic ones and zeroes traipsing down the wire. In the middle I have a shoebox full of logic circuits that read the input line and then tell the golfball what steps to do. Wait a minute! Turkey's mind was now racing. If I knew what parallel signals the solenoids on the printer mechanism need to print out various characters, I could work back through the logic to find out what kind of serial characters make the logic produce those codes. And, as a matter of fact, I already have those parallel codes in ol' Don Lancaster's book!"

Turkey jumped out of bed and started working feverishly. Using the photocopied diagrams that he'd received with his machine, he traced logic lines down one page and up another. He tore sheet after sheet off his roll of Scot towels, writing copiously on one, then another. Three a.m., four, five a.m., six – Turkey worked through the night. At last, as the sun peeked through the potted plants in his window, Turkey stood back in admiration of a neatly printed table. "There, if I'm not mistaken," he wept, "is the elusive serial code that'll make Mdm. Novar sing!"

"I think you've got it!" the Wizard said when later in the day they conferred. The Wizard sat down at Turkey's keyboard and ran a few tests, feeding the code out a character at a time using appropriate routines residing in Turkey's monitor. The Novar worked just fine, but still there was a thoughtful look on the Wizard's face.

"I'm not quite sure why this is working," he said as he examined the text hammered out by the Novar. "You have a Motorola MC 6850 Asynchronous Communication Interface Adapter on your System Monitor board that can be programmed to put out all kinds of character configurations with different numbers of start and stop bits, odd, even or no parity bits, and all that. But it's designed to handle seven or eight bit characters. The Novar's looking for six-bit characters. I've put a 15H into the instruction register of the 6859 which tells it to put out one start bit, an eight-bit character with no parity and one stop bit. Apparently the Novar's reading the first six bits of the characters we're putting through, reading the seventh bit as parity and ignoring the rest. So I can include the parity bit right into the eight-bit character stored in the lookup table of the conversion program, tell the ACIA ~~to put~~ out an eight-bit character and not bother about parity, and so far as the Novar is concerned it shouldn't matter a hill of beans. You might say we'll be operating on borrowed bits."

"You can't knock success."

"No, I guess not," said the Wizard. "As long as it works. . ."

And so dear reader, here ends the tale of Techno Turkey's journey through the underworld. The following Sunday the Wizard delivered a machine language program that fit just fine in spare RAM starting at FF00 on Turkey's System Monitor Board. By changing a single vector located at F815 from C3 C1 F9 to C3 00 FF and entering a few monitor commands to initialize both the 6850 interface chip and the Novar, Turkey was able to print out beautiful Selectric copy. In fact, Turkey loaned me his system to edit and type this very saga.

MORAL: Even a turkey can beat the high priests of technology at their own game – given courage, determination, and a little help from a wizard. ◻

Techno Turkey and the Haunted House

BY LLOYD R. PRENTICE

"There was neither moon nor stars," said Techno Turkey. "The sky was black with clouds. We shuddered in our coats and pulled them tightly around us as the wind howled and the thunder rumbled. Up on the hill we could see the house – the looming house – silhouetted against the horizon as lightning crackled across the sky."

Not bad, thought Turkey with a smirk as he looked down at the rapt little faces riveted up at him.

"And then what happened, Uncle Turkey?" asked the youngest urchin.

"You know, dummy," snapped urchin number two. "Now he's gonna do the shtick about Monster Mire."

"What a drag," said Turkey's precocious niece who thought she was Farrah Fawcett Majors, "Tell us about Dr. Evildoom and the wicked witch –"

"That's boring! I wanna hear the cave routine –" said a neighbor kid.

"Cave routine, schmrave routine," said the eldest urchin. "I'm gonna watch *Saturday Night Live* instead."

"Yaa! Yaa! *Saturday Night Live!*" they all exclaimed.

"Guess you're gonna have to touch up the old act," said Mrs. Turkey as the din of stampeding feet faded into the TV room.

"Act, schmract," said Turkey as he shuffled off to his basement study for a rousing round of Lunar Lander on the computer.

Now it so happens that in Turkey's house there is a dark creaky stairwell that leads down to Turkey's computer domain. Turkey had meant for some time to fix the light, but a certain problem getting his computer to talk to his Selectric terminal had kept him rather preoccupied. (See "Techno Turkey his Electric Selectric", September *PC*.) Now, as Turkey groped down these stairs with goblins on his mind, his face brushed a cobweb, which startled him; his sudden shift in weight triggered a creaky tread, which made him jump even higher; and as he came down his foot kicked a toy car which clattered down into the darkness.

Needless to say, with all this, Turkey's heart sprang up into his wattle, did a double gainer and arced back into his chest again. And the adrenaline was still surging as he made his way into the study where the faintly glowing pilot light on his computer began to sooth him.

Turkey turned on the video monitor and all of a sudden it struck him. Of course! A real haunted house! As real, that is, as modern technology can make it! He turned over an old envelope and started scratching feverishly and he scratched well into the night.

"Parallel ports!" thought Turkey. "I need a flock of 'em." As Turkey poured through his system documentation he discovered that his TDL System Monitor Board had two input/output ports, but both seemed to be tied up with one thing or another.

"How is my computer going to tell the skulking skeletons and galloping ghosts what to do without a spare port or two?" thought Turkey, not that he was all that sure what a parallel port did or how it did it. But Turkey jotted down task number one anyway: Add more ports to my system.

"Now, assuming I have the ports I need, how am I going to transform the feeble 5-volt logic signals that these ports put out into righteous control of real electrical power that'll drive my clanking chains and eerie lights? Some kind of relay, I suppose," Turkey thought. So task number two was to work out a system that would allow his computer to control honest-to-goodness 110V AC house power.

"Moreover," Turkey thought, "I not only want the ghouls and goblins in my house to do things, I want them to *respond* to things – like the little gobblers who visit the house – and for this I need various kinds of people sensors." So task number three was to figure out some simple sensing devices that would tell his computer when to spring its blood-curdling surprises.

"And, finally," Turkey thought, "I'll need some clever software to turn all these bits and pieces into a genuine horror show."

Turkey looked at his computer with renewed respect and shivered with glee at the bone-chilling effects that were racing through his mind. He'd fix their jaded little wagons good, he thought. Yes, indeedy, he would!

Next day, Turkey went to the computer store.

"Have I got a system for you!" said the man behind the counter when Turkey explained what he had in mind. "Look at this! It's got 2K of ROM and 2K of RAM and its got one purpose in life – controlling electrically powered devices!"

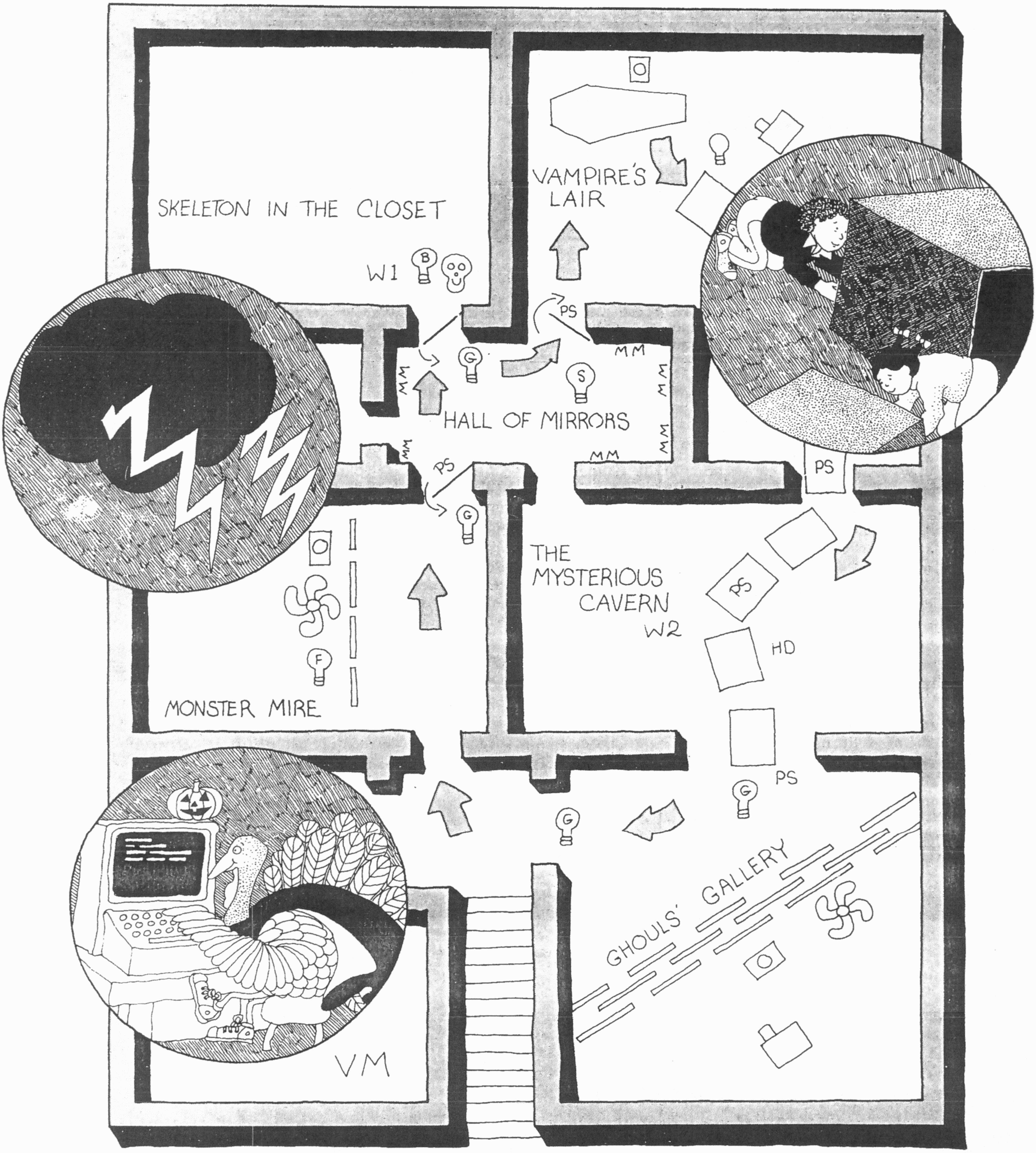
"No kidding," said Turkey.

"Yessiree! This little gem will control up to 100 electrical devices. You can tell it what time of day, what day of the month and what month of the year you want each device to turn on. You can tell it how long to let each device run. And built-in batteries save the program for up to four hours in case of power failure."

"Sounds good," said Turkey. "What's it called?"

"The Coby I."

"How much?"



G -	guide light	B -	black light	✤ -	fan	MM-	milar mirror	⬡ -	casket
F -	*flood light*	VM -	video monitor	⊡ -	speaker	☠ -	skeleton	⊏ -	projector
S -	strobe light	PS -	people sensor	¦¦ } -	cleesecloth scrim	W1/2-	whizzers 1 & 2	☐ -	refrigerator boxes

Illustration by Penny Carter

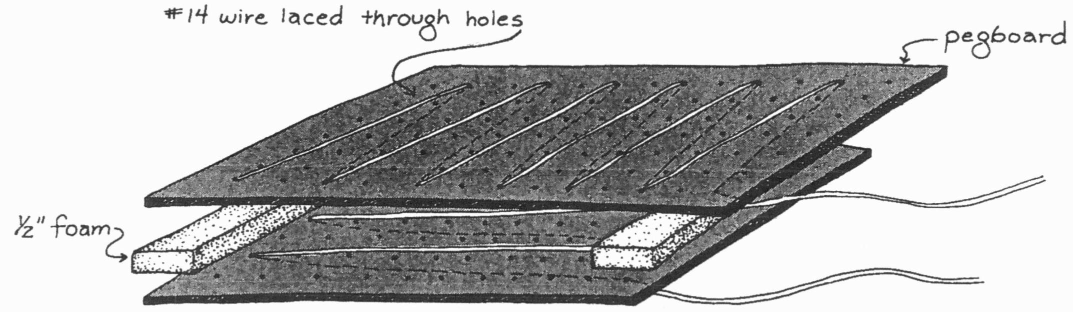


Figure 1

People detector – cut pegboard approx. 18" square – lace #14 wire through holes in opposite directions – glue pegboard and foam separaters into a sandwich – cover with a throw rug. When someone steps on it, wires on opposite pieces of pegboard touch, completing a circuit.

"Only $600."

"I see – "

"And each remote power control unit costs only $60."

Turkey gulped.

"How many devices did you want to control?"

"About 20 – "

"So you can get that haunted house of yours off the ground for only . . . " The salesman whipped out a pocket calculator, " . . . only $1800!"

"Little out of my range," said Turkey tersely.

"Well then, what about Mountain Hardware?"

"What's that?"

"Just one of the niftiest computer accessories on the market today! Plugs right into your S100 bus. And listen to this: it puts the control signals needed to control electrical outlets out through the existing AC lines, so you don't have to string a bit of wire. You can control up to 64 independent 500-watt AC receptacles."

"And how much does that cost?"

"The main controller board's only $329."

"Not bad," said Turkey.

"And each Dual Channel AC Remote is going to set you back $99."

And once again Turkey's hopes were dashed.

"Still too much, huh?"

The big bird nodded his head.

"Well I suppose you could use one of the relay boards like the Mullen or the Gimix and kludge something together . . . "

"What are those?" Turkey asked.

"They're S100 boards with a bunch of relays mounted on them. But I don't have any information."

"I don't know," said Turkey. "I'm about as handy as a . . . well, as a chicken when it comes to wiring things up."

"I guess I just can't help ya, fellow," said the salesman as he punched up a Pong game on the Apple II sitting on the counter.

Turkey was seriously considering giving up the whole haunted house insanity altogether when he thought of an old friend, Mylo, who reputedly could do with copper wire and electrical dohickies what Michaelangelo did with paint.

"No sweat, Turk Old Buddy," said Mylo when Turkey explained the problem. "Think of that old electricity flowing through your house wiring as a raging torrent of a river, except in this case the current is rushing first up stream and then down again 60 times a second.

"So what you've got to do is dam up the old flow – stick yourself a dam in the wire that your computer can open up or close down anytime it wants."

"And pray tell, how am I going to put a dam in my house wiring?" asked Turkey, thinking of Grand Coulee.

"Well, not exactly in your house wiring. You make yourself up an extension cord, see; you know, like plugs in-

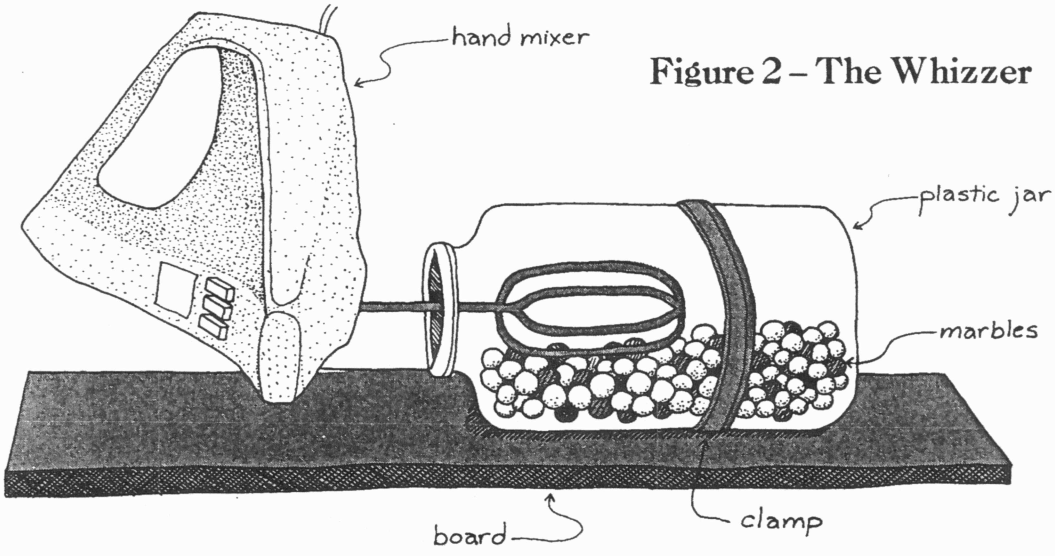


Figure 2 – The Whizzer

Figure illustrations by Susan Howard

to the wall. Then you put your dam in one wire of the extension cord."

"It's been a terrible day, Mylo," said Turkey, holding his head. "How do I dam up an extension cord?"

"With one of these little beauties." Mylo held up a small black thingamabob with three wire leads sticking out.

"I think I'm getting a headache," said Turkey.

"Called a triac," Mylo continued. "Take your extension cord, see, and then cut one of the wires up near the female receptacle – " Mylo whipped his wire cutters and with a deft flick snipped into a cord snaked across his workbench. "Better be sure everything's unplugged, of course, or you'll be one barbequed bird before you're done. But, anyway, now you got yourself two bare wires, see. Connect one wire to this here lead, MT 1, and the other to this here lead, MT 2." Mylo's soldering iron sizzled as he made the connection. (See Fig. 11.)

"And what about that third wire sticking out of the triac?"

"That's the trigger. That's how your computer stops or starts the flow through the dam."

Now Turkey began to see the light. "I see! I see!" he said excitedly. "Just plug that lead into the computer and you can turn the triac off and on, right?"

"Whoa, not so fast, Old Buzzard. It's not quite that simple. You got a few little details to work out."

"Like what?" Turkey asked suspiciously.

"Well, you said you were going to be turning electric motors on and off, right?"

"Right."

"Well, when you turn on an electric motor you get a heck of a magnetic field built up around the coils of the motor."

"So?"

"So, when you turn off that motor the magnetic field collapses and sends one big nasty jolt of electricity down the wire."

"And?"

"And, sometimes, it just might be too much for your triac to handle. Might just turn it right back on – something like a flash flood breaking down the dam."

"Just what I need," said Turkey. "A dam that can't turn off electric motors. Got any other good ideas, Mylo?"

"*Not so fast*: give me time. To fix this situation you just stick a resistor and a capacitor in series across the main terminals of the triac."

"Now I know I'm getting a headache," said Turkey. "What do they do?"

"They soak up any inductive kick that comes screaming down the wire when you turn off a motor."

"I see. Like using a giant sponge to catch the overflow of the dam, right?"

"Sort of – "

"So how do I hook this thing up to my computer?"

"When you solder one end of a 100-ohm resistor to Main Terminal 2, see – " Mylo pursed his lips as he completed the connection, clipped a test lead in place, plugged his electric drill into the extension cord and plugged the extension cord into the wall. Turkey noticed that he always kept one hand behind his back when he was working near a live circuit.

"Now, when I connect the free end of the resistor to the trigger lead of the triac, voila!" The workshop filled with the roar of the drill. When he removed the test lead, silence reigned.

"Wonderful, Mylo," said Turkey. "All my computer has to do is reach out and connect up the test lead every time it wants to turn on the triac, right?"

"No, as a matter of fact, there's a better way."

"And how's that?"

"You close the contact with a relay."

"A relay! Mylo, this thing's getting more and more complicated all the time!"

"So, whoever told you there was a free lunch?"

"Yeah, but why didn't we just use a relay in the first place – skip the triac and all that other bother?"

"You could, but then you'd need a big old expensive industrial relay that draws even more current and starts putting out inductive pulses of its own and you'd have nasty contact arcing and – uh! It depresses me to think about it!"

"OK, so we use a small el cheapo relay. How does my computer turn it on or off?"

"A relay driver. How else?"

"Great! Maybe we can get Paul Newman! Oy – Do I need an Excedrin," said Turkey.

"We could control the relay with a single transistor if we wanted to, but I've got something else in mind."

"I can't wait."

"Look, you've still got a few problems here, the way I see it. You've got to drive the relay, which takes a fair amount of current – at least, more than your port can supply.

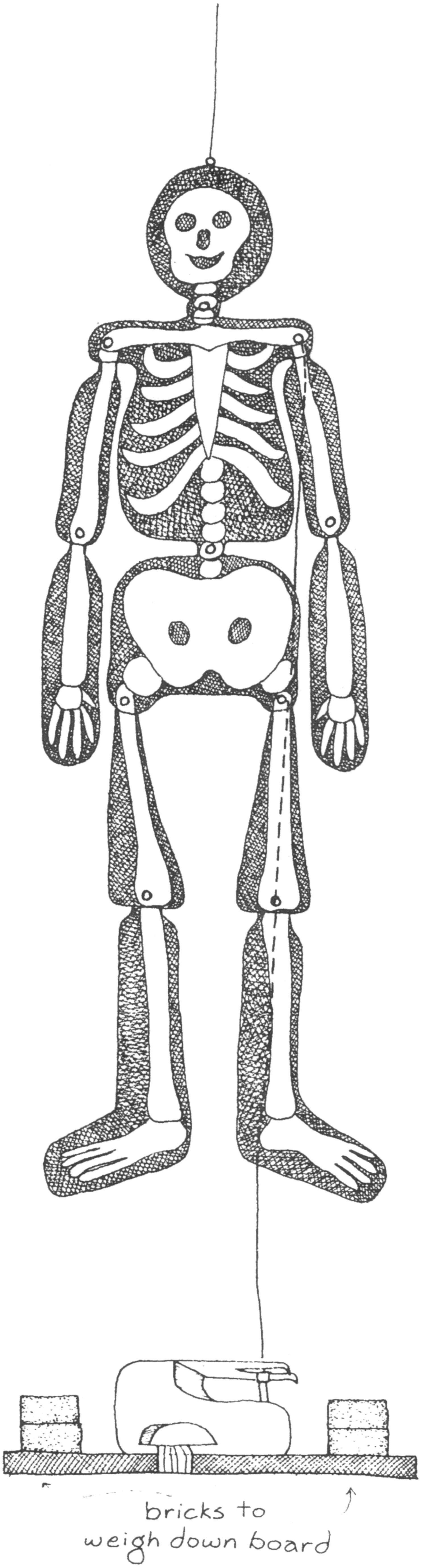

bricks to weigh down board

Figure 3

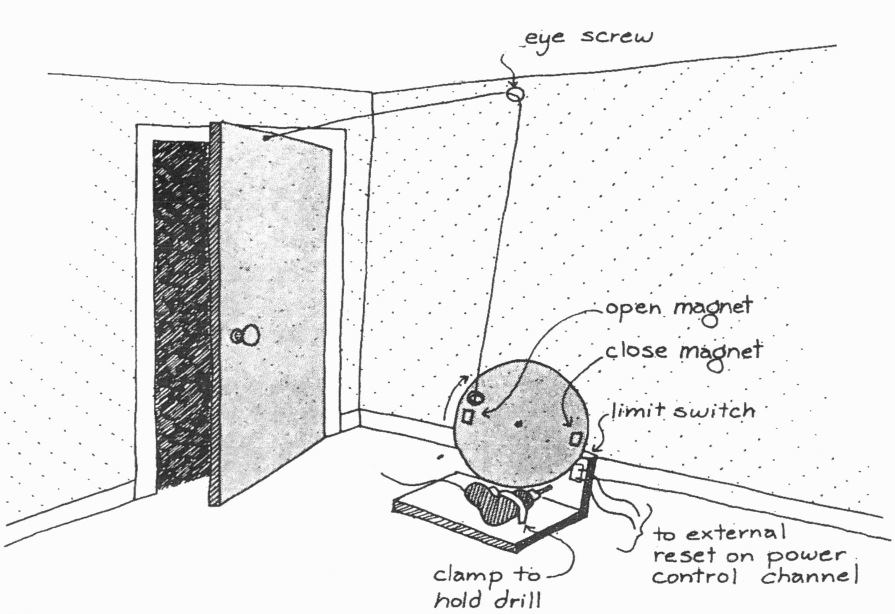


Figure 4

Door opening device – when computer turns on the electric drill, the disk turns. When the string is pulled to the bottom of the disk, the door opens. As the string revolves to the top of the disk, the door can be pulled closed by a weight on a string running through an eye screw in the sill.

You probably want to latch the relay so it stays on while the computer's off doing other things."(See Fig. 10.) "And you need to decode the signals from the output port so that –"

"Wait! Wait one minute! What do you mean 'decode the signals from the output port'?"

"You want to control a lot of circuits, right?"

"Right."

"If you latch the relay that controls each circuit, it's going to take two bits to control each one – one bit to turn it on and another turn it off. But you've only got eight bits per parallel output port. So unless you decode you're going to need five ports to control twenty circuits when you really only need one to do the job." (See Fig. 9.)

"You mean I'll only need one output port?"

"That's all."

"Now you're talkin'! How do we get away with that?"

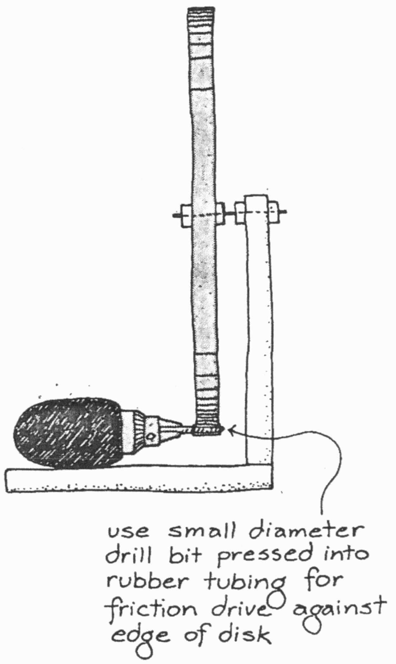
"We decode the eight bits coming out of the port the same way the computer decodes the address bus to access all its memory locations."

Turkey was looking morose.

"What's the matter, Old Bird?"

"There's a lot more to this than I expected."

"Well maybe. But it's not really so bad. I think we can handle each relay with just a few components and we need only one chip to decode up to four relays. Not only that, but I'm gonna throw in a big bonus."


use small diameter drill bit pressed into rubber tubing for friction drive against edge of disk

Detail of Figure 4

"What's that?"

"These power control units may end up some distance from the computer. When you try to run logic lines over a long wire you pick up noise problems something fierce."

"Terrific," said Turkey.

"I'm gonna throw in a line receiver that's the essence of simplicity –"

"I'm telling you my head's going to fall off, Mylo! What's a line receiver?"

"A circuit that accepts the valid logic signals coming down a long line, but tells any noise that comes along to go to blazes." (See Fig. 8.)

"Really, Mylo. This is quite beyond me –"

"Turkey, poor Cluck, tell you what I'm going to do: I'm going to design this Power Control System I'm talking about so you can buy every component you need at Radio Shack. You buy the parts and I'll wire up the first unit for you. We'll build them on a nifty little PC board that Radio Shack sells, so all you've got to do is drill a few holes, solder a few wires and you've got yourself a dandy Power Control Unit. Of course, if you want twenty of 'em, it'll be more work, but you can do it like an assembly line."

"No kidding!" said Turkey. "How much do you think it'll all cost?"

"You're paying premium prices at Radio Shack, but if you want to buy the convenience, I'd say . . . maybe less than $10 per controlled function."

"That means the electronics for the whole haunted house would come to around $200."

"If you shop for parts, you might get it down to $150. And not only that, after Halloween you can use the system to control anything you want around the house. That computer of yours just might start paying its keep."

"About time!" said Turkey, nodding assent. "About time that electron guzzler does some real work for a change!"

When Turkey left Mylo's he was quite excited. First thing, he visited all his friends and neighbors to scrounge some items he needed.

"Mrs. Webster, do you have an old electric mixer I could borrow for a few weeks? And how about a hair dryer?"

"Say, Claude, could I borrow your electric hand drill?"

"You've got a few old cassette players around? Wonderful!"

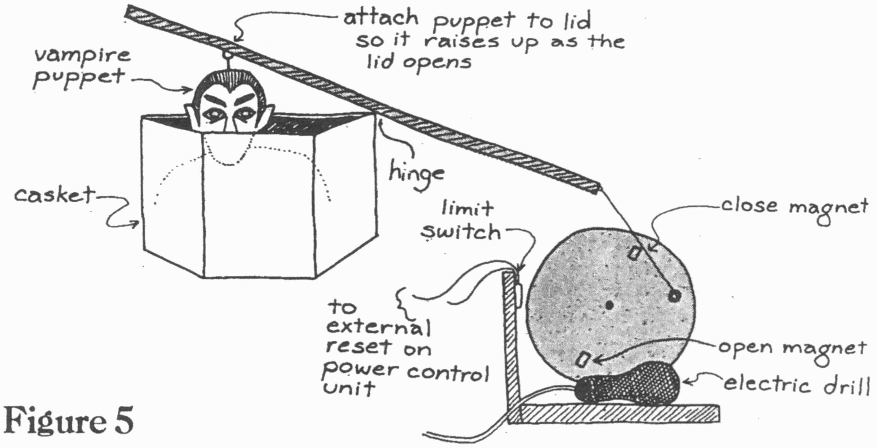
And he was absolutely ecstatic when a friend who managed a rock group loaned him a black light and a disco-type strobe.

For the next few weeks Turkey buried himself in his basement workshop.

"What's the old man up to?" his kids whispered among themselves.

"Don't know. Everytime he looks at me he giggles like a fiend."

"I'm not sure I like this one!" the littlest Turkey said.



attach puppet to lid so it raises up as the lid opens
vampire puppet
casket
hinge
limit switch
close magnet
to external reset on power control unit
open magnet
electric drill

Figure 5

Device for opening lid to casket

Turkey made one more trip to the computer store to buy a parallel output board. Nearly everything was outside his price range, but he did find a super little kit by Solid State Music called an I/O & S-100 Proto-typing Card which gave him one parallel input port and one parallel output, plus large prototyping areas to add more ports if needed. While the concept of the board was terrific, he had to spend a whole evening at Mylo's figuring out how to put it together since the instructions were about as meaningful as the random memory patterns that showed up on his video monitor before he loaded a program into his computer.

Turkey also consulted two other VIPs. He talked the Wizard who'd helped him with his Selectric terminal into writing some software, and then he talked to Mrs. Turkey.

"You're going to do *what* to the house?" Mrs. Turkey exclaimed.

"It's only for one night –"

"Well I'm going to Mother's!"

"And miss all the fun?"

"It'll take me a week to dig out of the mess –"

"But look, when it's all over we can use the computer for the security system we need. And you know how the kids never turn out the lights when they leave a room? The computer can do it and save on our electricity bill! And listen to this –"

"I can't wait –"

"I can set this thing up so the computer can sense when the garden needs water and turn on the sprinklers automatically!"

"Alright already! But I've got one condition."

"What's that?"

"I get to paint the skeleton!"

(*Author's note:* What follows is quite frankly a fantasy. For some strange reason, magazine editors require Halloween stories around the Fourth of July. Since Turkey was having enough trouble getting his sparklers to sizzle, there was no way he could

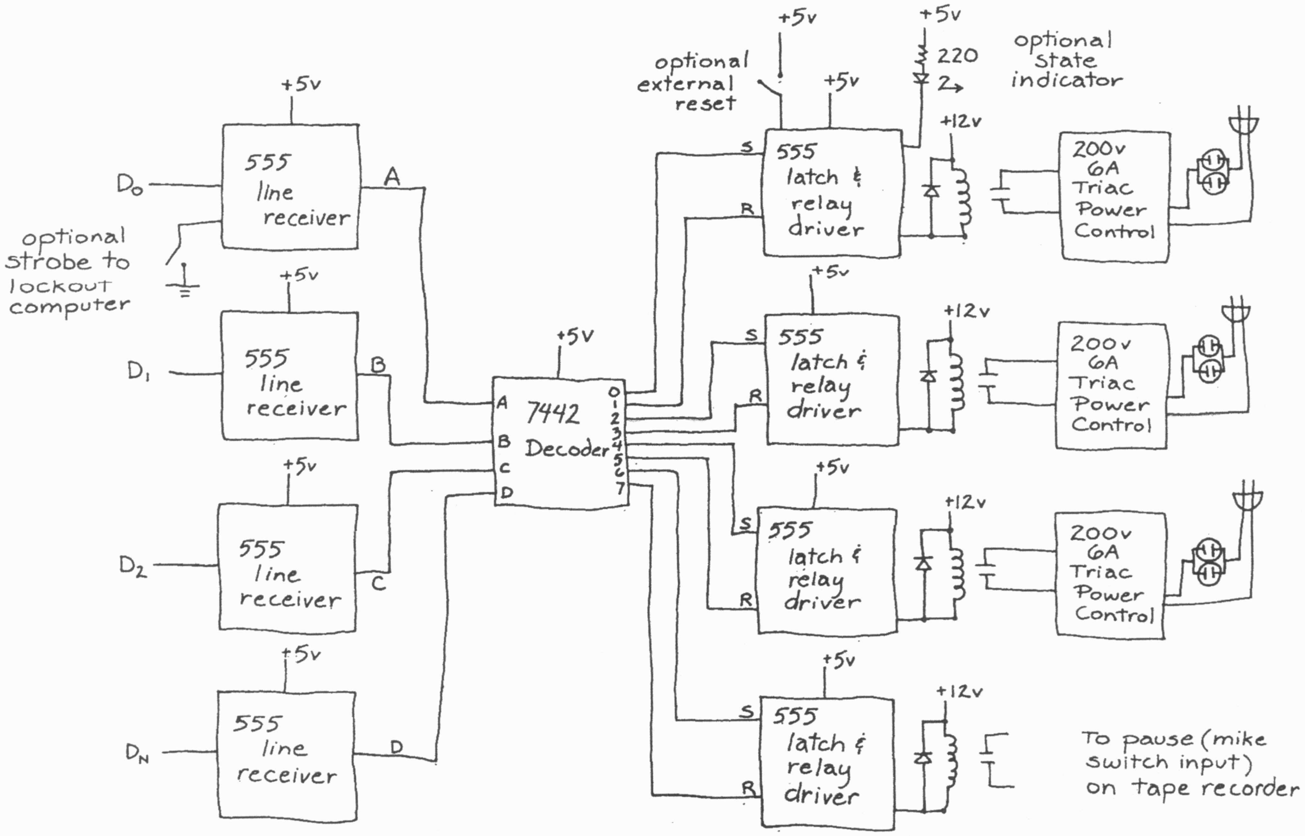


Figure 6

This shows how basic circuit modules are organized in one power control box. Haunted House requires five identical boxes – giving a total of 15 controlled 110V AC receptacles and 5 controlled low-current relay closures.

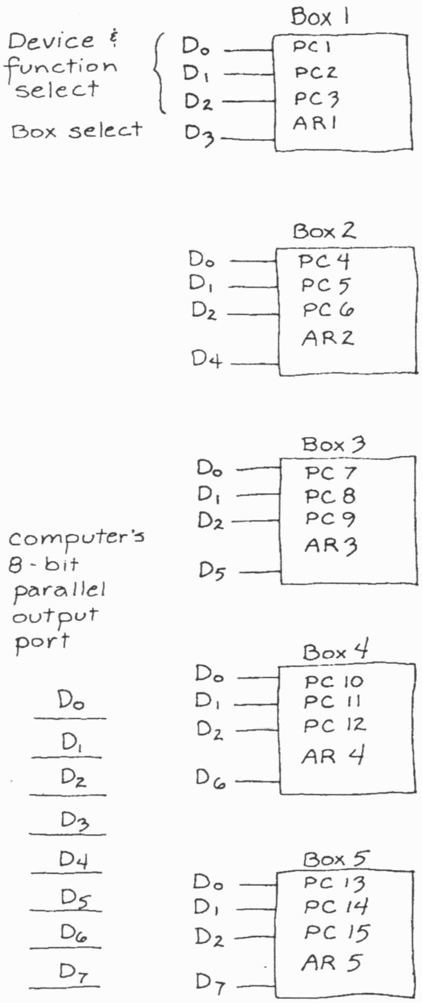


Figure 7

This diagram shows how the five power control boxes are wired up to the computer. In each box there are three 110V AC power controlled outlets and one low-current relay used to control the pause function (mike switch) on a tape recorder.

get the whole haunted house in place three months early. But Turkey schemed and dreamed anyway and gave us exclusive permission to share his innermost fantasies.)

The moon was a gigantic pumpkin on Halloween night. Turkey lurked just inside his front door, costumed appropriately as a Transylvanian count in silken tails. When the doorbell rang he ushered in the greedy little trick-or-treaters and invited them on a tour of his old family estate.

Mrs. Turkey, dressed like an ancient family retainer with a hunchback, accompanied them through the rickety old wrought iron gate that led into . . . Monster Mire!

Turkey then sat at his video monitor and traced the progress of the party through the terrifying inner sanctums. Satisfying it was, and even more satisfying were the delighted screams that issued from within and the flushed, excited children who asked to go through again and again.

In Monster Mire they meet a vicious storm with wind and thunder and lightning. They feel the mud ooze under their feet (they are asked to enter the estate barefoot) and spiders and creepy vines brush their faces. But most of all they hear the wolves – the baying of hungry wolves growing closer and closer. And then, just in the nick of time, they see the old house – dark, silent, looming and ominous on the hill. The door creaks open and a spooky voice invites them in . . . "Come in! Come into my house! The wolves are hungry tonight. Come into my house! It's dangerous in the Mire!" And they enter . . . The Hall of Mirrors!

Now the door creaks closed behind them. It's dark and eerie. Suddenly lights flash and strange shapes loom around them. Hideous laughter fills the air. Just as they think they're going mad, there's darkness and silence again. They try a door only to confront . . . The Skeleton in the Closet.

When they open the door, a skeleton glows in the dark. At first it's still and then suddenly jumps to life with a nerve-shattering rattle of bones. They slam the door and try another only to find themselves in . . . The Vampire's Lair.

In this room dim candles sit on a table. A faint red light flicks on and reveals a casket in the middle of the room! At that very moment the casket begins to creak open and a dark form with leering red eyes rises up out of

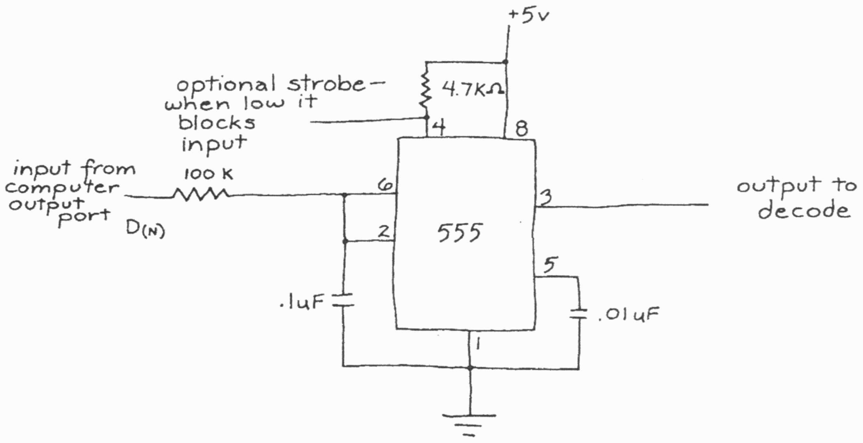

+5v
4.7KΩ
optional strobe – when low it blocks input
4
8
input from computer output port
D(N)
100 K
6
2
555
3
output to decode
5
.1uF
.01uF
1

Figure 8

Line receiver – This circuit is quite insensitive to noise that builds up when the power control box is a long distance from the computer. The optional strobe input can be used to lock out computer control at the box itself – just run it through a switch to ground. Approximate cost including pc board – $1.50.

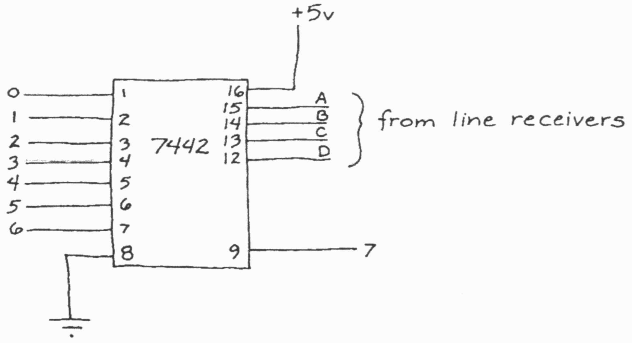


Figure 9

Decoder circuits – one needed to decode the on and off functions for four circuits within one power control box. Approx. cost including pc board – $1.

the box. Dracula himself! The room is plunged into darkness and they hear a hideous scream! Just in the nick of time a faint light and a soft voice beckons them into . . . The Mysterious Cavern.

It's dark in the cave. There are slimy, cold things on the floor and cobwebs brush their faces. Ghastly rattles come from the walls and a spooky voice bounces around in the cave. A strange gust of wind blasts them in the face just before they emerge into . . . The Ghoul's Gallery!

Here the air is filled with shimmering ghosts; luminous spirits; dark, contorted faces; monsters of every description. Just before their hearts come to a complete stop, a deep booming voice addresses them:

"You have passed unharmed through the most wicked ordeals devised by the Minds of Evil. Your hearts are Good, your souls are Pure. Go now! Go into the world of light and let no Shadow of Evil pass before you."

"This is terrific, Turk!" says Everet Mastermeyer, his neighbor. "Listen, I belong to the Jaycees and we do one of these every year down at the firehouse as a fund raiser. How about giving us a hand next year?"

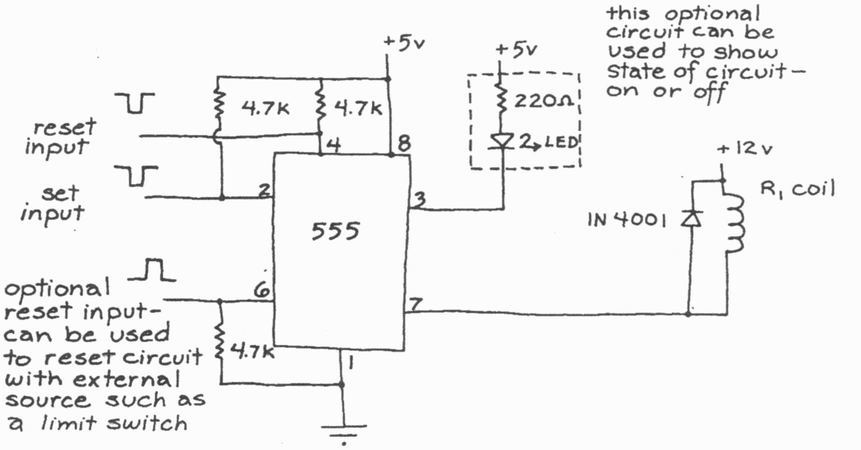

this optional circuit can be used to show state of circuit – on or off
+5v
+5v
4.7K
4.7K
220Ω
2 LED
+12v
reset input
set input
4
8
2
3
R1 coil
IN 4001
555
optional reset input – can be used to reset circuit with external source such as a limit switch
6
7
4.7K
1

Figure 10

Latch and relay drives – when set input is brought from +5V to ground the relay turns on and stays on. When the Reset input is brought from +5V to ground, the relay turns off and stays off. Note optional LED state indicator and optional reset control. Approximate cost of circuit including pc board and relay < $4.00.

"I was really scared, Daddy!" said the youngest Turkey. "Did you do all of that with your computer?"

"Yeah, I even forgot to watch *Laverne and Shirley!*" said eldest urchin. "I'll get all my friends over to see this thing."

"What're you and your computer planning next?" asked Mrs. Turkey.

Turkey just smiled. Wait 'til Christmas, he thought to himself. Even old St. Nick is going to have the time of his life!

How it works

Most of the effects in the Haunted House are created around common audio-visual devices, household appliances and power tools. These items are turned on and off by the computer through

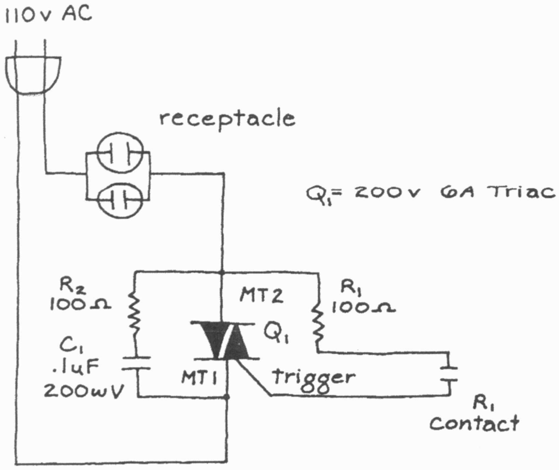
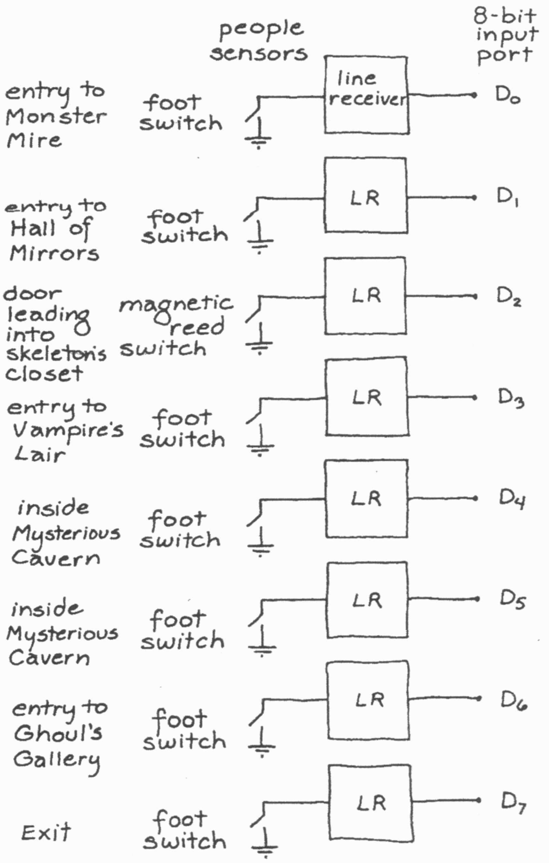


Figure 11 △

Triac power control circuit – When relay contact closes, triac turns on, providing power to the 110V receptacle. R₁ limits current in the triac trigger circuit and across the relay contact. R₂ and C₁ absorb large voltage surges that occur when motors are turned off. Approximate cost of circuit including pc board and receptacle <$3.

Figure 12 ▷

Sensor circuits. See Fig. 7 for Line Receiver Circuit.



five identical power-control boxes. (See Fig. 7.) Each box contains four independent control circuits – three 110V AC receptacles that may be independently switched on or off by the computer and one controlled relay, also computer controlled, that may be used to switch low-level direct current such as that found in a tape recorder pause control circuit. Four simple circuit modules are interconnected in each control channel. (See Fig. 6.) All of the components used in the modules are available from Radio Shack. While sensitive 12V relays were used in my prototype (required 10 ma for closure), any relay that does not exceed the voltage and current-handling capacity of the 555 IC may be substituted (15V at 100 ma max.). Power for the circuit modules in my prototype was provided by a power supply built up on the 5-Volt Regulated Power Supply Project Board sold by Radio Shack. Note in Fig. 6 that the optional strobe input to each line receiver module may be used to lock out computer control. This feature is useful when you want to work on a particular effect while the computer program is running. The optional external reset line into the latch/relay driver circuit is used to turn off the control channel independently of the computer. This feature is used in both the door opening effect and the casket opening effect. Limit switches detect the position of the disk in its cycle and stop rotation at the proper position. (See Fig. 4,5.)

Table 1 – Software Control and Variable Assignments

Control variables in program	Data codes to box on	off	Box number	Control unit	Controlled device	Effect
D(1) T(1) T1(1)	0FH	0EH	1	PC1	floodlight	Mount floodlight behind cheesecloth scrim in Monster Mire. Prop leaves and twigs in front of floodlight to cast eerie shadows on scrim. Program will flash floodlight at random intervals to simulate lightning. To add to the effect, put a water-soaked foam mattress pad on the floor. Use plastic drop cloth under the pad to protect the floor. All visitors should go through the house barefoot, of course. Hang soft twigs, leaves, threads, rubber creepy-crawlies from the ceiling over the path through Monster Mire.
D(2) T(2) T1(2)	0DH	0CH	1	PC2	fan	Fan causes cheesecloth scrim to ripple and creates wind effect in Monster Mire.
D(3) T(3) T1(3)	0BH	0AH	1	PC3	night light and electric drill that powers door opener	Toward the end of period in Monster Mire, a small night light goes on to guide attention to the door of the Haunted House. At the same time, the door starts to open, powered by the electric drill.
D(4) T(4) T1(4)	09H	08H	1	AR1	Relay controlling pause function (mike switch on tape recorder 1)	This tape unit has sounds of thunder and howling wolves on it. If you can't find a continuous cassette, you'll have to repeat the basic sound effects several times on a longer tape and rewind frequently between visitors to the house. Most of the audio effects work best on continuous cassettes.
D(5)	17H	16H	2	PC4	strobe light	Walls of the Hall of Mirrors are lined with mirrorized mylar film available from plastics supply houses. Unwrinkled aluminum foil will work in a pinch. Strobe light goes on when people sensor 2 is actuated, indicating a visitor has entered. A hangman hanging from the ceiling might add to the spookiness.
D(6) T(6) T1(6)	15H	14H	2	PC5	night light	At end of the Hall of Mirrors effect, a small night light comes on to illuminate the door leading to Skeleton in the Closet. On the door is a sign that says "OPEN ME".
D(7) T(7) T1(7)	13H	12H	2	PC6	blacklight	When the door is opened into the Skeleton's closet, a blacklight comes on to cause the luminous paint on a skeleton puppet to glow.
D(8) T(8) T1(8)	11H	10H	2	AR1	Relay controlling pause function on tape recorder 2	This tape unit has creaking door effect for door leading from Monster Mire into Hall of Mirrors. It also has creepy voice inviting visitors into the house – and hideous laughter that goes with the strobe light.
D(9) T(9) T1(9)	27H	26H	3	PC7	sabre saw electric mixer	Sabre saw controls movement of the skeleton puppet. See Fig. 3. The saw comes on about 15 seconds after the door into the Skeleton's Closet opens. At the same time, whizzer 1, powered by the electric mixer, comes on.
D(10) T(10) T1(10)	25H	24H	3	PC8	night light	This light guides visitor to door leading into Vampire's Lair. It comes on at end of Skeleton routine.
D(11) T(11) T1(11)	23H	22H	3	PC9	projector	A slide projector with deep red filter over the lens casts eerie light on casket in Vampire's Lair.

The optional LED state indicator lets you know whether a particular control channel is on or off.

Each power control box is connected to an eight-bit parallel output port through four data lines. Each of the five high-order bits coming from the port go to a separate box. These bits select which box receives a given control command. The three low-order bits go to all five power control boxes. These select which controlled function is turned on or off. Actual control commands are listed in Table 1.

Movement of people through the Haunted House is detected by a series of eight people sensors. These are simple switches, either activated when stepped on, or, in one instance, activated when a door is opened. (See Fig. 1.) These switches connect to an eight-bit input port through simple line receiver circuits to minimize the chance of false triggering due to noise picked up by the long cables connecting the sensor switches to the computer. (See Fig. 8.)

Every half second or so the computer checks the input port to see if a sensor has been activated. For every output control channel N (where N = 1 to 20) there are three associated timing variables in the Haunted House program. The D(N) variable determines the length of time between actuation of the input sensor and execution of control function N. If the control function is off and is supposed to be turned on at the end of the D(N) delay, then the T(N) variable deter-

Table 1 – Software Control and Variable Assignments

Control variables in program	Data codes to box on	off	Box number	Control unit	Controlled device	Effect
D(12) T(12) T1(12)	21H	20H	3	AR3	Relay controlling pause function on tape recorder 3	This tape has creepy sounds associated with casket opening in Vampire's Lair. At end of segment is a bloodcurdling scream.
D(13) T(13) T1(13)	47H	46H	4	PC10	electric drill	The electric drill powers the casket opening device. It comes on about 10 seconds after visitors enters Vampire's Lair. The vampire dummy is hung from the inside of the casket lid so the dummy rises as the lid opens. Use light-emitting diodes to make the dummy's fiery eyes. A limit switch on the casket opening device actuates the manual reset switch on PC10 to stop the electric drill when the casket is fully open. When the visitors trip the people sensor at the entrance to the Mysterious Cavern, the computer program closes the casket to prepare for the next visitor.
D(14) T(14) T1(14)	45H	44H	4	PC11	night light	This light guides visitors into the Mysterious Cavern. It goes on at the end of the Vampire effects.
D(15) T(15) T1(15)	43H	42H	4	PC12	electric drill	A stiff piece of rope about 8" long is mounted in the chuck of the drill. The drill is clamped to a support outside one of the boxes used for the Mysterious Cavern. (The cavern itself is made of refrigerator or other large appliances boxes.) When the drill goes on, triggered by the people sensor at the entrance to the cavern, the rope whaps against the side of the box with nerve-shattering effect on anyone inside the box.
D(16) T(16) T1(16)	41H	40H	4	AR4	Relay controlling pause function on tape recorder 4.	This tape unit plays all the sound effects that go with the Cavern routine.
D(17) T(17) T1(17)	87H	86H	5	PC13	electric hair dryer	When the second people sensor in the Cavern is actuated, the electric hair dryer goes on, blasting hot air into the cave through a hole in the box – ideally about face level. Incidentally, the cave can be made even creepier by hanging threads from the tops of the boxes, sprinkling cooked spaghetti on the floor (feels like worms when you step on it barefoot) and putting one or two plastic bags filled with ice on the floor of the cavern – eyuck!
D(18) T(18) T1(18)	85H	84H	5	PC14	fan and slide projector	Hang cheesecloth scrims from the ceiling. Fan is aimed to make cheesecloth ripple. Projector is set on automatic slide advance to project ghosts, witches, ghoulish faces and so forth on the three scrims. Since the images are projecting through one cheesecloth onto another, there will be three images hanging in space for each slide.
D(19) T(19) T1(19)	83H	82H	5	PC15	night light	This light guides visitors out of the Haunted House at the end of the Ghoul's Gallery routine.
D(20) T(20) T1(20)	81H	80H	5	AR5	Relay controlling pause function on tape recorder 5	This relay provides audio for Ghoul's Gallery – chains clanking, groans and voice at the end.

mines the length of time the function will stay on once executed. If the function is on and is supposed to be turned off at the end of the D(N) delay, then the T1(N) variable determines the length of time the function will stay off once executed. Using both the T(N) and T1(N) variables in the output routine lets the control function flash on and off repeatedly.

Actual timing of the system will depend upon the clock speed of your computer and the speed of your particular BASIC. Thus, the timing values shown in the program are nominal and will have to be adjusted to fit your system. A value of 1 assigned to the D, T or T1 variable corresponds to one cycle through the main loop of the program. In my system, this cycle corresponded to about 0.5 sec. Thus, if D(3) = 20, say, then power control channel 3, which is controlled by D(3) according to Table 1, will be actuated approximately 10 seconds after people sensor 1 is activated.

Table 1 also shows how the different effects are produced. Wind effects are produced by window fans; lightning is produced by flashing a floodlight on and off quickly at random intervals. Sound effects are recorded on continuous-loop cassette tapes (available from Lafayette).

Several special effects are shown in Fig. 2 through 5.

Safety should be carefully considered in every phase of the Haunted House design and layout. All electromechanical devices should be well enclosed or otherwise kept out of the path through the house. Wiring should be strung very carefully so visitors don't accidentally trip. All electrical circuits should be enclosed and kept away from sinks, toilets, bathtubs or other watery environments where electrical shock is a possibility. In other words, when the house is set up, go through it several times looking for potential hazards and eliminate them.

Program Listing

BY PETER HENRY

```
10 E=0
20 REM ****** TECHNO TURKEY'S HAUNTED HOUSE CONTROLLER ****
30 REM ***********************************************************
40 REM WRITTEN BY          PETER HENRY     JULY, 1978
50 REM ***********************************************************
60 REM THIS PROGRAM MAY BE MODIFIED FOR OTHER APPLICATIONS
70 REM WHERE A TIME (OR UNTIMED) RESPONSE TO RANDOM (I.E.
80 REM HUMAN) STIMULUS IS TO BE RESPONDED TO.
90 REM ***********************************************************
100 DIM D(20),T(20),F(20),T1(20),P(8)
110 02=1
120 REM 02 IS AN OPTION FLAG THAT, WHEN SET TO 1, CAUSES ~~NO~~ A
130 REM CHECK TO BE MADE TO SEE IF A PORT IS ON OR OFF WHEN
140 REM IT IS TURNED OFF OR ON (RESPECTIVELY).  THIS SLOWS
150 REM THE PROGRAM, BUT MAKES IT MORE RELIABLE...........
160 FOR A=1 TO 8
170 P(A)=0
180 REM P IS THE SET OF PEOPLE SENSOR FLAGS, WHICH HELP TO
190 REM DETERMINE IF A PERSON IS JUST STANDING ON A PEOPLE
200 REM SENSOR, IN WHICH CASE THE SIGNAL IS IGNORED...
210 NEXT A
220 REM T IS TIME TO OFF CONSTANT
230 REM T1 IS TIME TO ON CONSTANT
240 REM D HOLDS THE TIME CONSTANTS FOR EACH OUTPUT, AS WELL
250 REM AS THE FUNCTION TO BE PERFORMED ON 'TIMEOUT'.  THIS
260 REM WORKS AS FOLLOWS............
270 REM EACH D VARIABLE IS INITIALIZED TO 0 AT THE START OF THE
280 REM PROGRAM.  THE MAIN LOOP OF THE PROGRAM DECREMENTS EACH
290 REM D VARIABLE BY ONE.  WHEN A VARIABLE GETS DECREMENTED
300 REM BELOW 0, THE FUNCTION IS PERFORMED FOR THAT DEVICE.
310 REM   THE FUNCTION IS DERIVED BY TESTING THE VARIABLE TO
320 REM SEE IF IT EQUALS ITS INTEGER PART.  IF IT DOES, THE
330 REM DEVICE IS TURNED OFF.  IF THE NUMBER
340 REM IS NOT AN INTEGER, HOWEVER, THE DEVICE IS TURNED ON.
350 REM      THEN 'TIMER'  CONSTANT IS LOADED INTO THE D VAR-
360 REM -IABLE FROM THE CORRESPONDING T VARIABLE.  THIS
370 REM CONSTANT IS FOR PURPOSES OF TURNING A DEVICE ON
380 REM (OR OFF IF IT WAS JUST TURNED ON) AFTER A CERTAIN
390 REM TIME DELAY.  FINALLY, WHENEVER A DEVICE IS TURNED ON
400 REM OR OFF, THE CORRESPONDING FLAG (F) VARIABLE IS TURNED
410 REM TO 1 OR 0.   THIS IS NOT REALLY NEEDED, BUT IT
420 REM HELPS TO SPEED UP THE PROGRAM.
430 FOR A=1 TO 20
440 D(A)=0
450 T(A)=0
460 T1(A)=99999999
470 F(A)=1
480 NEXT A
490 REM ***** THIS IS THE MAIN LOOP
500 I=80
510 O=80
520 REM ***THE ABOVE ARE THE INPUT AND OUTPUT PORT NUMBERS...
530 REM THEY MAY HAVE TO BE CHANGED FOR DIFFERENT SYTEMS.
540 FOR G=1 TO 20
550 D(G)=D(G)-1
560 IF D(G)>0 THEN 610
570 A=G
580 PRINT "UNIT #";A;" HAS TIMED OUT."
590 IF D(A)=INT(D(A)) THEN GOSUB 1100 ELSE GOSUB 1320
600 REM TURN IT OFF OR ON DEPENDING ON INT() RESULT
610 NEXT G
620 PRINT "MARK...";
630 PRINT
640 REM SAMPLE THE PEOPLE SENSORS...
650 S=INP(I)
660 REM THIS BINARY BREAKDOWN ROUTINE COULD BE MORE EFFICIENT
670 REM IN TERMS OF THE PROGRAM BEING OPTIMIZED, BUT A LIST
680 REM OF 8 IF STATEMENTS IS THE FASTEST WAY POSSIBLE ON
690 REM MOST MICROCOMPUTERS TO BREAK A BYTE INTO 8 BITS.
700 IF S<128 THEN P(1)=0
710 IF S<128 THEN 750
720 IF P(1)=1 THEN 750 ELSE P(1)=1
730 GOSUB 1600
740 S=S-128
750 IF S<64 THEN P(2)=0
760 IF S<64 THEN 800
770 IF P<2)=1 THEN 800 ELSE P(2)=1
780 GOSUB 1890
790 S=S-64
800 IF S<32 THEN P(3)=0
810 IF S<32 THEN 850
820 IF P<3)=1 THEN 850 ELSE P(3)=1
830 GOSUB 2070
840 S=S-32
850 IF S<16 THEN P(4)=0
860 IF S<16 THEN 900
870 IF P(4)=1 THEN 900 ELSE P(4)=1
880 GOSUB 2200
890 S=S-16
900 IF S<8 THEN P(5)=0
910 IF S<8 THEN 950
920 IF P(5)=1 THEN 950 ELSE P(5)=1
930 GOSUB 2430
940 S=S-8
950 IF S<4 THEN P(6)=0
960 IF S<4 THEN 1000
970 IF P(6)=1 THEN 1000 ELSE P(6)=1
980 GOSUB 2500
990 S=S-4
1000 IF S<2 THEN P(7)=0
1010 IF S<2 THEN 1050
1020 IF P(7)=1 THEN 1050 ELSE P(7)=1
1030 GOSUB 2580
1040 S=S-2
1050 IF S=0 THEN P(8)=0
1060 IF S=0 THEN 1090
1070 IF P(8)=1 THEN 1090 ELSE P(8)=1
1080 GOSUB 2700
1090 GOTO 540
1100 REM TURN OFF THE DEVICE ROUTINE...
1110 IF 02=1 THEN 1160
1120 IF F(A)=1 THEN 1150
1130 PRINT "DEVICE STATUS SAYS: DEVICE ALREADY OFF!"
1140 GOTO 1310
1150 REM DETERMINE THE BOX #...
1160 B1=INT((A-1)/4)+1
1170 REM DETERMINE THE LINE #...
1180 L=A-(B1-1)*4
1190 PRINT "DEVICE IS DESIGNATED: BOX";B1;" LINE";L
1200 REM ASSEMBLE THE COMMAND, AND OUTPUT IT...
1210 C1=1
1220 FOR X=1 TO B1+2
1230 C1=C1+C1
1240 NEXT X
1250 C1=C1+(L-1)*2
1260 OUT O,C1
1270 PRINT "DEVICE";A;" TURNED OFF..."
1280 PRINT
1290 D(A)=INT(T1(A))+.5
1300 F(A)=0
1310 RETURN
1320 REM TURN ON DEVICE ROUTINE...
1330 IF 02=1 THEN 1380
1340 IF F(A)=0 THEN 1370
1350 PRINT "DEVICE STATUS SAYS: DEVICE ALREADY ON!"
1360 GOTO 1530
1370 REM DETERMINE BOX #
1380 B1=INT((A-1)/4)+1
1390 REM DETERMINE THE LINE #
1400 L=A-(B1-1)*4
1410 PRINT "DEVICE IS DESIGNATED: BOX";B1;" LINE";L
1420 REM ASSEMBLE COMMAND...
1430 C1=1
1440 FOR X=1 TO B1+2
1450 C1=C1+C1
1460 NEXT X
1470 C1=C1+(L-1)*2+1
1480 OUT O,C1
1490 PRINT "DEVICE";A;" TURNED ON..."
1500 PRINT
1510 D(A)=INT(T(A))
1520 F(A)=1
1530 RETURN
1540 REM EACH PEOPLE SENSOR ROUTINE IS CUSTOMIZED TO DO
1550 REM SOME SPECIFIC ACTION WHICH IS TO BE DONE AFTER
1560 REM A GIVEN PEOPLE SENSOR HAS TRIGGERED...........
1570 REM THEREFORE, IN THIS ARTICLE, THE ROUTINES HAVE
1580 REM BEEN WRITTEN TO CORRESPOND TO THE SCENARIO OF
1590 REM TECHNO TURKEY'S HAUNTED HOUSE.................
1600 PRINT "PEOPLE SENSOR 1 TRIGGERED!"
```

TLC
the Educator's Guide to Personal Computing
teaching • learning • computing

DECEMBER 1983

VOL. 1, NO. 2 $3.00

A Microcomputer Survival Kit

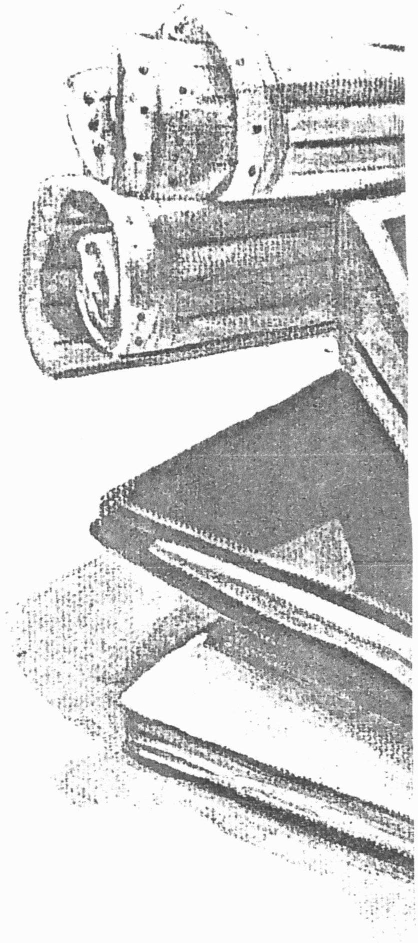
Setting Up A Computer Lab

Trouble In Logoland

Computer Lab Contest Details p. 73

A microcomputer

Ideas and resources to help survive the day the computer arrives.



Suddenly having a computer in your classroom is a bit like suddenly having your child's fiancee as a houseguest. With so much getting-acquainted to do, where do you begin? Fortunately, getting to know your computer is a pretty straightforward task. You can learn everything you need to know to make a computer useful in less than 20 minutes. Of course, you can also go on learning for the rest of your life. We'll leave the rest of your life up to you. But in the 20 minutes it will take to read this article, you can be on the road to computer literacy.

To begin with, let's look at a couple of computing myths. Two disparate myths often characterize the new user's view of computing. One, the Genius Theory of Computing, holds that if you can't get into MIT, you can't tame a computer. Equally debilitating is The Myth of Computer as Santa Claus. Knowledge and experience don't matter, this myth holds, because computers are magic. Just let a computer down the chimney (well, through the classroom door) and reap bountiful gifts of learning.

Neither is true, of course. What is true is that the computer is a tool — a powerful tool, but nothing more than a tool. This is good news for classroom teachers. It means that you already know the most important thing about computers in your classroom: your own instructional goals.

Whether you want to help remedial students improve reading scores, unleash the creative potential of your gifted students, or prepare youngsters for the computerized world they will enter, the computer is a tool to help you do it. The same keen judgment and educational savvy with which you evaluate slidetapes, textbooks or films will guide you as you assess this newest teaching aid. *Your* methods, *your* philosphy, *your* students will shape the way you use computers in the classroom.

This is not to say that you don't need special information — you do. But the starting point, as with any curriculum decision, is the question closest to you: what and how should my children learn? To formulate reasonable answers to this question, you need to understand what computers can do.

The silicon soul of this new machine is as prosaic as sand. It has no insight; no values; just the ability to do three things supremely well. It can store, retrieve and manipulate information. Information is anything — words, pictures, numbers, sounds, temperature, pressure or light — that can be translated into a form the computer understands.

The computer's native tongue is a language comprised of ones and zeros called *binary codes.* What happens when you communicate with a computer is rather simple. You enter information through an *input device.* Keyboards are the most common input device; the joysticks with which kids play video games run a close second. The input device translates incoming information into binary code and sends the message off to the *central processor.*

The central processing unit (often referred to as the CPU) is the brain of your computer — an electronic switching station that accepts and executes instructions. You can think of the central processor as a dumb but fast and efficient clerk who fetches an instruction and carries it out — millions of times a second, without error or fatigue.

Our efficient clerk would overload his circuits without someplace in which to store information and from which to retrieve it. Enter *internal memory.* You can think of internal memory as a long bank of file folders.

Laurie Beckelman and Lloyd R. Prentice are, respectively, editorial director and president of Prentice Associates Inc., a Boston-based company that produces educational software and computer publications.

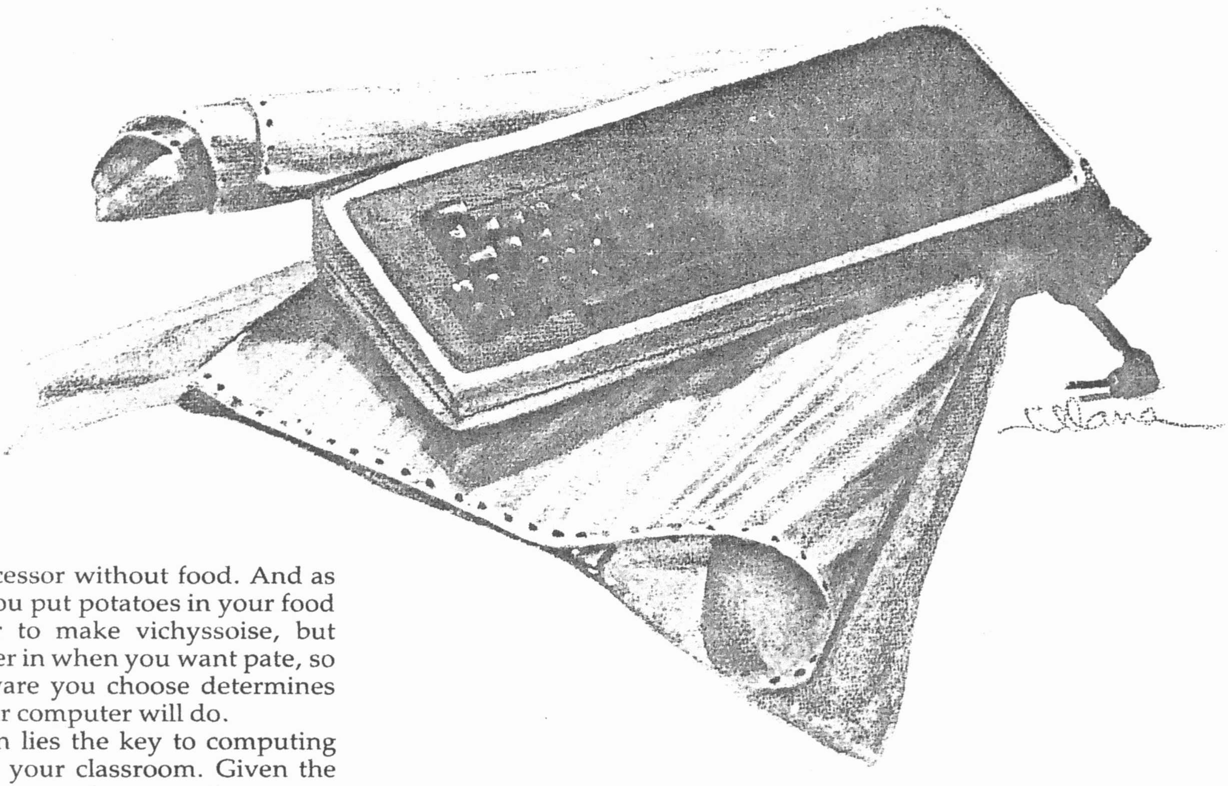
survival kit

by Laurie Beckelman and Lloyd B. Prentice

Each folder has an individual label, or address, so that the central processor knows what's stored where. One "folder" is one *memory location*. It is a row of eight, electronic, on-off switches. Each switch is called a zero or a one. Eight bits equal one *byte* — the amount of memory necessary to store a single character — an *a* or a *g*; an ⋆ or a *1*.

Once the computer has faithfully manipulated information as instructed, it needs a way of showing you the results. An *output device* transforms the pattern of switches the computer understands into information you understand. Video display screens — either monitors or television sets — are output devices. So are printers. But computer output is not necessarily visual. Sound and voice synthesizers are output devices that produce music or speech, respectively. The robot arm that tightens bolts on an assembly line is also an output device, albeit a very specialized one. Given the right input and output devices, a computer can do anything from play a fuge to print report cards to tell you when your car is low on gas.

But input and output devices aren't enough. A third element is essential to this alchemy: *software* — the instructions, or *programs,* that tell the computer *(hardware)* what to do. Hardware without software is like a



food processor without food. And as sure as you put potatoes in your food processor to make vichyssoise, but plunk liver in when you want pate, so the software you choose determines what your computer will do.

Herein lies the key to computing power in your classroom. Given the appropriate software, the same machine that brings you *Pac-Man* can become a spelling tutor or a problem-solving aid, a vehicle for artistic expression or the heart of a science lab. That new machine in your classroom has more expressions than Marcel Marceau. Versatility is its greatest virtue.

The subject of software brings us back to our central question: what should my students learn with computers and how? Educators currently use computers in the classroom in three major ways: to teach about computing; to teach traditional subjects; and to manage instruction. Instructional management is beyond the scope of this article, but let's examine how computers help teachers teach.

Teaching about computing

Teaching about computing falls under the rubric of computer literacy. Just what is computer literacy? That depends upon whom you ask.

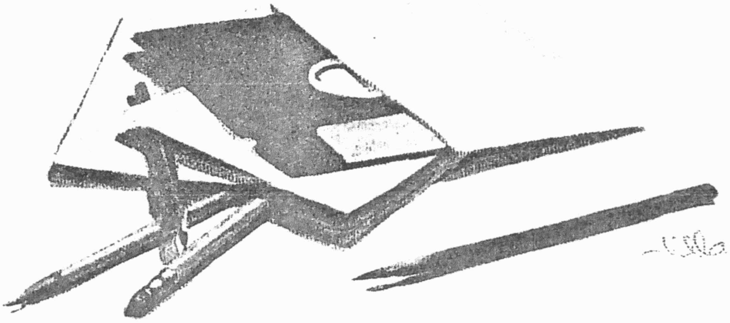
To some educators, a child is computer literate once he or she can define byte, Basic, disk and data, plus tell you that banks, airports, scientists and tax collectors all use computers. According to this school of thought, knowing what a computer is and how it's used in society constitutes literacy. But others belive that the road to *computer literacy* is paved with programming commands: one is forever illiterate until fluent in Basic, Pascal or another computer tongue.

And for yet a third group, both these definitions are inadequate. This group holds that computer literacy not only encompasses programming and a general understanding of computers, but also implies familiarity and ease with the technology that is deep enough to make computers tools of daily work and living. By this definition, computer literacy is to a computer-oriented society what reading and writing are to a print-oriented society.

Which of these definitions you accept and the degree to which you can structure curriculum to meet the goals it implies, will, of course, depend upon your teaching style and students. We personally agree with the broadest definition and believe that courses in computer literacy should cover the following:

How computers work and why they are useful. Most students don't need a detailed technical understanding of how computers work, but they do need to understand that computers are tools with limitations. They need to know what computers can do and what they can't; how they do what they do; and, most important, how computer and human potential differ.

How to use the computer as a tool. Only by mastering the computer can a student fully understand its strengths and limitations. Mastery means learning to *program,* to write computer software. Computers are polyglot machines. Basic, Cobol, Fortran, Logo, Pascal, Forth, Ada, C — all are computer languages. When someone



learns to program, he or she learns one of these languages and all the rules, or syntax, that govern its use.

The languages you will most likely encounter are Basic, Pascal and Logo. Basic, which stands for Beginner's All-purpose Symbolic Information Code, is resident in many microcomputers. It is frequently the first language today's programming student learns. In some schools, however, Pascal is replacing Basic as the teaching language of choice. Pascal is a newer, more structured and, say some, better language. The Educational Testing Service has created considerable controversy by choosing Pascal for its new advanced-placement exam in computer programming.

Logo completes the triumvirate. It is a language developed especially for education; one that children as young as five have learned to use. It teaches programming through intuitively obvious commands and graphics that give children immediate feedback on their work.

How to use computers responsibly. With stories of computer pranks making the cover of *Newsweek,* our need to teach children the ethics of computing is clear. But using computers legally is only one side of using computers responsibly. We must also arm our youngsters with the values and attitudes that will best help them confront the significant choices, opportunities and problems they will face as the result of computer technology.

> *"Your methods, your philosophy, your students will shape the way you use computers in the classroom."*

How computers will shape our future. Students need to understand the computerized society in which they will work and live. Through such an understanding alone can they assess the opportunities before them and the job skills they will need for success. Students should also be aware of the next wave of technology — robots and artificial intelligence that are no longer the stuff of science fiction.

Many excellent books and software packages can help you plan a computer literacy curriculum for your students. Some of these are included in the Compact Survival Kit (page 38). For a more thorough listing, we recommend *Instructor's Fall 1983 Computer Directory for Schools (Instructor Magazine,* 757 Third Avenue, New York, N.Y. 10017). This is a complete directory of products serving the educational computing field.

Teaching with Computers

Not all computer-using educators are teaching computer literacy, of course. Many are using computers to support more traditional academic goals. The *Instructor* directory lists close to 2,000 educational software products. Programs offer everything from sophisticated economic simulations to straight spelling drills; they cover art, social studies, physics, math, music, language arts and health — and just about anything else you can name. Not surprisingly, they also reflect a variety of pedagogical approaches. Here again, your personal approach to teaching will determine what type of software will work best for you. Let's take a broad look at the major categories of educational software.

Drill-and-practice. The underlying assumption of drill-and-practice is that some things must be learned the hard way — by rote memorization. And the fastest way to memorization is repetition and rehearsal. The computer is a whiz at offering drill-and-practice. It never gets impatient or tired. It always asks the question in the same tone. Further, it can provide drills in the guise of games, adding motivational spice to the bland porridge of addition drills or spelling bees.

Tutorial. Add instructional material to our humble drill-and-practice program, and it suddenly steps up in the world. It is now a tutorial, in which we present the pupil with new information and then test his or her comprehension. If our student demonstrates comprehension, the program branches to a positive response; if not, it branches to a negative response.

We can embellish a tutorial in many ways. We can use elaborate graphics to introduce the concept; we can provide review and remediation if the student answers incorrectly; we can record right and wrong answers and so track a student's progress.

Tutorials are a classic form of **computer-aided instruction** (CAI) and, indeed, are what many people think of when they think of CAI. The Computer Curriculum Corporation of California and Control Data Corporation in Minnesota, among others, have developed extensive tutorial systems, spanning many grades and subjects.

> *"Some of the best games...provide concrete metaphors for thinking about abstract problems."*

Tutorials and their paler cousins, drill-and-practice programs, are lauded by some and disdained by others. Proponents point to studies that show increased learning among both children and adults who have worked with tutorials; opponents point out that such programs tie youngsters into lock-step curriculums, thereby discouraging creative thinking and dehumanizing the learning process. This controversy points out once again the importance of evaluating educational software in light of your own teaching philosophy and needs.

Games. The lessons of the video arcades have not been lost on computer educators. Kids whose school-day attention spans are timed to commercial breaks develop the patience of lion trainers at the game arcades. While not all teachers agree, those who favor educational computer games say, "If we need sound, color and fast action to turn kids on to learning, sound, color and fast action it will be."

Games offer two major educational benefits. A well-designed game sharpens the player's concentration and problem-solving skills. Moreover, games can help the teacher present difficult or tedious material in an interesting way. Some of the best games offer still a third benefit — they provide concrete metaphors for thinking about more abstract problems.

Three game formats are popular today. The shoot-or-be-shot *Space In-*

(continued on page 41)

vaders clones are fast, involving and require good hand-eye coordination. Lightening Software's *MasterType* uses this format to reinforce touch typing skills — type the right letter fast enough and you blast an alien. *Multiploy,* from Reston Publishing, uses a similar approach to drill math.

Maze games a la *Pac-Man* are another genre. We developed a game called *Chambers of Vocab* for Reader's Digest. In *Chambers,* vocabulary words slither like beasts through a maze. The player "conquers" the beasts by matching each word with its correct definition.

Still a third popular format is based upon a computer game called *Adventure.* The object of an adventure is to search for treasure in a labyrinth of caves or rooms while, at the same time, avoiding various hazards. Tom Snyder of the Learning Connection turned this concept to good educational use in a series of games he developed for McGraw Hill. Called *The Search Series,* these games promote critical thinking skills while teaching social studies and science concepts.

Simulations. One of the most exciting types of educational software is the simulation — a program that emulates the logic of a physical or social system. Computer simulations are usually interactive. They allow the learner to change a variable and observe the consequence to the system as a whole. One of the earliest educational simulations was a program called *Lemonade,* produced by the Minnesotta Educational Computing Consortium (MECC). In *Lemonade,* students manipulate such variables as price per glass and amount of advertising in an effort to make a lemonade stand pay off.

> *"That new machine in your classroom has more expressions than Marcel Marceau."*

Tools of expression. The computer's reputation as master of rigorous logic and systematic thinking often eclipses its prodigious ability to encourage creative expression. The reason computers are fine tools for artistic expression is that they let the creator "mess around." Ill-begot phrases, notes or strokes vanish at the touch of a key or a brush of the light pen (a special input device often used for entering drawings into the computer).

Increasingly powerful software tools are available in this area. Among them: *The Bank Street Writer,* a simple word processor distributed by Scholastic Inc.; *Paint,* a program from Reston Publishing that lets users do exactly what it says (with computer "oils," of course); *Story Machine,* from Spinnaker Software, which lets youngsters develop stories that the computer then animates; and *Music Maker,* a music composition module from Texas Instruments.

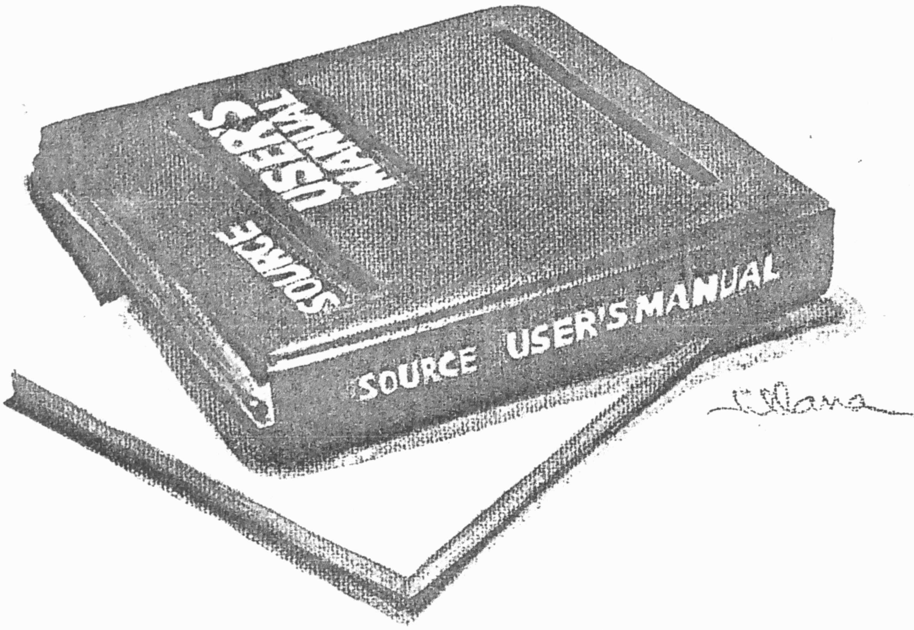
Programming languages. We've already talked about programming languages under computer literacy, but one final word here. Languages serve a dual function. They not only enable the user to master the computer but also foster better problem-solving skills. Indeed, many educators believe that the computer's finest contribution to education is its potential to enhance problem-solving ability in children.

These educators argue that programming gives concrete expression to the abstract principles of problem solving. In so doing, it makes these concepts easier to grasp and gives children a new way of thinking. The logical, systematic approach to problems learned while programming, they say, will transfer to other areas of learning.

Now that you're on the road, here are five next-steps to keep you going:

1. Define your instructional goals. Think deeply about what computers can do for you.
2. Locate and evaluate software that can help you meet your goals.
3. Once you've found appropriate software, develop a concrete plan for integrating computers into the rest of your curriculum.
4. Accept the fact that you'll have students who will learn faster than you. That's fine. Use them as a resource to tutor other kids or to write computer programs that the rest of the class can use. Remember, the key to using computers effectively is not technical wizardry, but making best use of the resources you have.
5. Keep up with this fast-moving, exciting field. The resources on page 38 will help you do it.

Simulations available today include Muse Software's *Three Mile Island,* in which the user runs a nuclear power plant; EduWare Services' *Rendezvous,* which simulates a space shuttle flight; EduTech's *Birdbreed,* a genetics program that simulates inheritance of color in parakeets; and a series of chemistry simulations from Radio Shack. As the above examples demonstrate, one advantage of simulations is that they let the learner experiment with systems that would otherwise be inaccessible.



Class of '82

Educational software is getting better—but don't trash the textbooks yet. Producers still have a lot to learn.

By Lloyd R. Prentice

Current statistics from credible sources are probably the best indicators of growth in the educational software field. But I have another metric. I visit my friend Rick.

Rick first called me some 18 months ago. "My company is thinking about distributing software to schools. Can you recommend products that we should add to our line?"

I was impressed. Rick's company had been selling stuff to schools for more than 100 years—everything from chalk erasers to shuttlecocks.

At that time Rick's office was tucked in the corner of a cavernous warehouse. We threaded down aisles of storage racks that groaned under pallets of blackboards and pencils and scissors and paste, and we entered a space the size of a roomy broom closet. "Our new microcomputer area," Rick announced proudly.

By my next visit the stroll to Rick's area was shorter and less of an adventure. Rick had a new assistant and new space. Their two desks seemed lost in the expanse of the new office.

During my third visit carpenters were hammering away in an adjacent area installing floor-to-ceiling shelves. "Storage for our inventory of software and books," Rick told me. He introduced me to his new secretary.

With each visit I watched the floor of Rick's domain disappear under desks and racks and computer stations. Rick introduced me to the manager of his new retail outlet and hinted at big deals brewing.

My last visit with Rick was just a few days ago and it's clear that lack of space is again crimping his style. My mind flashed back to the cavernous warehouse and for an instant I saw pallets of floppy disks and three-ring binders piled to the fluorescent fixtures high above my head.

There are both pessimists and optimists in the educational software field. One publisher told me that schools just don't have money for software. But Talmis, a respected market research firm, sees sales growing from near zip three years ago to some $75 million by 1985. Certainly my visits with Rick affirm that sales are on the rise. But the big question is—just what is being sold? What's available to help kids learn and how good is it? Progress in these areas is not so easy to gauge.

Recently my company did a detailed survey of the hardware and software products available to educators. We identified 1004 separate software products targeted for grades K through 12. This number, in fact, is misleading since many of the products that we turned up are actually series that include 10, 20 or 30 different units. So we're talking about maybe 1500 to 2000 separate programs. We found programs for art and music, computer literacy, early childhood, guidance, language arts and reading, library skills, mathematics, science, social science, special education and vocational and business education. We also found programs for instructional management and various "authoring" systems to make the process of developing instructional software less of a chore.

These 1004 packages were produced by some 217 companies—169 produce software exclusively, 30 produce both hardware and software and 18 produce both software and books. From the point of view of quantity, then, producers are tripping over one another to get the product out and they seem to be cranking out something for everyone.

But what about quality? How good is the stuff? Here we plunge into a bucket of worms.

There are two sides to the quality question in educational software—technical quality and pedagogy. Technical quality comes down to questions like these:

- Does it run?
- Is it easy to use?
- Are the screens clear, easy to read, appropriately illustrated?
- Is the branching logic correct?
- Is the documentation complete, easy to use and correct?

Pedagogy is somewhat stickier. Indeed, the issue of appropriate pedagogy is the Falkland Islands of education. One camp is looking for structured presentation of content, explicit instructional objectives, accountability and emphasis on basic skills. Another camp wants to create a rich environment of information and tools and leave the learning to the innate curiosity, creativity and "constructionist" instincts of the child. This camp believes that children "construct" their own knowledge at their own pace out of their personal experiences with the physical and social environment.

Clearly, your basic philosophy of education will influence your approach to software development and evaluation.

Lloyd R. Prentice is president of Prentice Associates, Inc., 46 St. Johns St., Boston, MA, a software development and consulting company.